# PCI S5335 Developer's Kit

# User Manual

**February 2005**

**Revision 1.0**

**Applied Micro Circuits Corporation**
**6290 Sequence Drive**
**San Diego, CA 92121-4358**
**http://www.amcc.com**

# Table of Contents

# Table of Contents (cont.)

Chapter 1

# Introduction

The AMCC PCI Developer's Kit contains everything needed for the PCI developer to immediately begin operating and experimenting with a S5335 based PCI design. For software engineers, the Developer's Kit is a fully functional PCI to Add-On bus test card. The programmer can immediately begin testing and operating numerous aspects of PCI Bus to Add-On Bus data transfers, timings, control and overall operation. The programmer can also test and become familiar with the various aspects of PCI BIOS functions and PCI Configuration Space operation. A Windows based application allows the user to view and change device register contents from the PCI Bus as well as view and change PCI configurations, including downloading, editing, configuring and programming capability to the optional serial boot load nvRAM contained on the main Developer Kit PCI card. A second Windows application allows the user to view and change device register contents from the AddOn Bus.

For the hardware designer, the Developer's Kit provides fully functional PCI to Add-On bus design examples. The main S5335 PCI card shows Add-On bus connection to onboard SRAM.

The Developer's Kit comes complete with schematics, PCB artwork, bill of material (BOM), CPLD equation source code, nvRAM source code, nvRAM tool, and application software.

## Developer's Kit Overview

The PCI Developer's kit contains one printed circuit board plus a CD-ROM with software tools. The S5335 PCI card contains an S5335, SRAM, a pre-programmed CPLD containing Add-On bus control functions and a 9 pin, RS-232 connector to communicate with the AddOn Bus. This card was developed to demonstrate interconnection of the S5335 PCI interface chip to the PCI Bus and interconnection of the S5335's Add-On Bus to a basic SRAM design. The onboard CPLD is specifically programmed to control the Add-On bus for Active Mode data transfers for burst or single cycle data reads and writes to the SRAM. It’s presently being controlled through the GUI and CPLD.

The Add-On bus signals are also routed to a set of external application connectors. These connectors provide the designer with additional Add-On bus connection capability. The designer can utilize these for attaching his/her own application PCB to the PCI card's Add-On bus. These connectors are designed to provide connection of the user's custom PCB or a logic analyzer.

It is important for the designer to remember, the developer's kit was designed to demonstrate various aspects of S5335 user design. The specific CPLD, Add-On logic components and software were chosen to support multiple application illustrations. Therefore, the device costs and complexity is more than will be necessary for many applications.

## Files on CD-ROM

Shown below are the basic directory folders for the CD-ROM supplied with the developer kit. The content description of each folder and sub-folder is also listed. For normal hardware development, it is only necessary to copy the utility '.exe' programs to the hard drive for easy access and execution. The CD-ROM contains the Utility programs for the S5335DK and the C source code, library and include folders for the programs. It also contains example h, library and include C files.

**Readme.txt** – Start here first.

**S5335 Hardware Folder**

- **Schematic** – OrCad v9.2.1 schematic files for the PCI card
- **Bill of Materials** – Parts list for the PCI card
- **PCB** – Gerber and PADS v5.0.1 PCB files
- **CPLD** - Altera CPLD code (Verilog) for the PCI card

**S5335 Software Folder**

- **Application Installation\PCI Side** - Contains the installation files for the AMCCPCI5335Win.Exe application.
- **Application Installation\AddOn Side** - Contains the installation files for the AMCCPCI5335AddOnWin.Exe application.
- **NVRAM Tool Application** – Contains the source (Source Code directory) and installation (Installation directory) files for the NVRAM utility program and drivers.
- **NVRAM Files** – Contains nvRAM files in Intel Hex format for burning the nvRAM on the S5335 Development board.
- **Driver Installation** - Contains the driver and INF files for installation of the drivers under the Windows 98 and Windows 2000 operating systems.

**S5335 Documentation Folder**

The S5335 data sheet (visit www.amcc.com for the latest revision)
The S5335 Developer Kit manual

# The Developer's Kit Goal

The S5335DK was designed to help both hardware and software engineers go into production with a new design as quickly as possible. Hence, AMCC has provided the following:

- A fully functional hardware design example of an SRAM interface.
- Documentation text files to help come up to speed quickly on all parts of the S5335DK.
- Hardware - all source files to re-create the S5335DK board and use them as the basis for your design. Also included the source file for the CPLD.
- Software - program examples and nvRAM utility tool source code to help develop your new software and debug hardware.

Win95, Win98, Win2000, WinNT, and Linux software device drivers for the S5335 are currently available through our Development Partners. Please visit our web site at www.amcc.com for up-to-date links.

# Developer's Kit Features

## *The S5335 PCI Card*

The Primary design aid to the Developer's Kit is the main PCI Developer Card. This board contains the S5335 device interfaced to the PCI Bus giving the developer a functional example of device location, trace lengths/routing and decoupling. The Add-On Bus of the S5335 is interfaced to board signal headers, SRAM and a CPLD device. The CPLD supplied serves as an example of Add-On Local Bus interface control to SRAM. There is also a 9 pin, RS-232 serial connector for communications with the AddOn Bus.

## *The Software*

For the software developer, the S5335DK provides a fully functional PCI bus to Add-On bus test environment. Much software development can start immediately, without waiting for your new hardware to be built. The software developer can become familiar with the PCI BIOS and PCI Configuration Space Registers. Additional software functionality supports downloading, editing, and programming data to an optional serial boot load nvRAM on the S5335DK. A second application is included to allow controlling the AddOn Bus operational registers through the 9-pin, RS-232 serial connector. Source code is included for a slimmed down version of the application that supports the functions of the nvRAM.

# Developer's Kit Contents

The S5335 Developer Kit contains the following hardware, software and documentation:

- Primary Developer S5335 PCI Card. This card is the main design aid of the S5335 Developer Kit. We recommend designers follow this design as an example of correct device location, trace lengths, routing and decoupling. The card consists of S5335, serial nvRAM, 32K DWORDs SRAM, pre-programmed CPLD, oscillator, RS-323 transceiver, and Add-On bus connectors.

- CD-ROM
  - Developer Software Tool
  - Schematics
  - Gerbers
  - Bill of Material (BOM)
  - nvRAM and CPLD source code
  - S5335 Data Sheet
  - Developer Kit User Manual (this document)
  - Install files for all Windows applications
  - Install files for the Windows software driver

- Mounting bracket and hardware

# Developer's Kit Software

The following software programs and applications are supplied on CD-ROM:

- AMCCPCI5335Win –PCI side Test and Diagnostic application
- AMCCPCI5335AddOnWin – AddOn side Test and Diagnostic application
- AMCCNVRAMWin - nvRAM Tool

The AMCCPCI5335Win application is supplied as an executable program and allows full access to the PCI side of the S5335 Development board. It includes the application and system driver. No source code is supplied for this application.

The AMCCPCI5335AddOnWin application is supplied as an executable program and allows full access to the AddOn side of the S5335 Development board. No source code is supplied for this application.

The AMCCNVRAMWin application is supplied as both an executable program and source code for modification and compiling by the user. This application allows access only to the functions required to read and write the nvRAM only and does not have access to any other S5335 functions.

These applications run under Window 98 and Windows 2000.

Also supplied on CD-ROM are nvRAM image files in Intel Hex format that can be loaded and burned into nvRAM of the S5335 Development board using either of the above applications.

## *Win98 and Win2000 Software Device Drivers*

Included on CD-ROM are the system driver files required to install the S5335 Development Board in a Windows 98 or Windows 2000 system. These drivers are supplied in executable format.

Included with the AMCCNVRAMWin source code are source files for a scaled down version of the device driver. Care should be taken to not install this version of the driver over the top of the driver used by the AMCCPCI5335Win application, since it does not support all the functions required.

To obtain full system drivers with source code, Applied Micro Circuits Corporation maintains links on its web site (www.amcc.com) to third party software companies having device drivers for the AMCC S5335 PCI device. AMCC works closely with vendors, making sure their products enhance your development process. However, we leave all aspects of development, marketing and support to these development partners. See the above web sites for the latest information on these device drivers.

## *Win95, WinNT, and Linux Software Device Drivers*

The applications contained on this CD-ROM are not supported under Windows 95, Windows NT, or Linux.

Currently Applied Micro Circuits Corporation maintains links on its web site (www.amcc.com) to third party software companies having device drivers for the AMCC S5933 and S5335 PCI devices. AMCC works closely with vendors, making sure their products enhance your development process. However, we leave all aspects of development, marketing and support to these development partners. See the above web sites for the latest information on these device drivers.

### *Software Tools for the AMCCNVRAMWin Program*

To modify and compile the AMCCNVRAMWin application, the following tools are required:

- Microsoft Visual C/C++ 6.0

To modify and compile the system device driver used with the AMCCNVRAMWin application, the following tools are required:

- Microsoft Visual C/C++ 6.0, or other appropriate C language compiler for use with the Device Driver Kit (DDK).
- Windows Device Driver Kit (DDK) for the target operating system

No compilers or DDK's are supplied on this CDROM.

## Notation Conventions

**Active-Low Signals**-Signals which are asserted (or active) in the low voltage state are defined with a trailing number/pound (#) sign within the schematics or with a leading exclamation (!) for CPLD equations.

The following designations are used throughout this manual when referring to the size of data objects.

A BYTE is an 8-bit object
A WORD is a 16-bit, or 2 byte object
A DWORD is a double word and is a 32-bit or 4-byte object.

Hexadecimal notations are indicated with a trailing "h" or a leading 0x.

9A4Fh
0110h

Binary notations are indicated with a trailing "b".

1010b
0110b

# PCI SIG ID Policy

The PCI Special Interest Group has developed a device and card identification system to ensure all PCI Bus devices are uniquely identified. This identification system allows software operating systems to load appropriate software drivers based on the ID numbers. Use the following table as a reference guide for temporary PCI identification numbers for use in the developer's kit. The indicated numbers are the factory defaults pre-programmed into the onboard nvRAM and are loaded into the S5335 PCI Configuration Registers during power-up initialization.

| Configuration Register | Name | Value |
|---|---|---|
| Vendor Identification | VID | 10E8h |
| Device Identification | DID | 5335h |
| Revision Identification | RID | 00h |
| Subsystem Vendor ID | SVID | 0000h (Not supported) |
| Subsystem Identification | SID | 0000h (Not supported) |

The PCI SIG has divided identification numbers into two groups. Group one is dedicated to the chip manufacturer to uniquely identify the silicon device on the PCI bus. Group two is dedicated to the end user or board manufacturer to uniquely identify the end product on the PCI bus.

- **VID** - The vendor identification number is assigned by the PCI SIG to the IC manufacturer. In this case, 10E8h has been registered to the name Applied Micro Circuits Corporation for identifying AMCC as a PCI chip device manufacturer.
- **DID** - The device identification number is assigned by AMCC under it's rights of VID assignment. AMCC assigns a unique DID to each of its PCI chip devices. In this case, AMCC has assigned 5335h to uniquely identify the S5335 PCI interface chip.
- **Revision** - The revision number is also assigned by AMCC. This number is assigned and programmed to identify the revision level of the silicon die within the device package. In this case, the register is hardwired to the silicon's revision.
- **SVID** - The sub-vendor identification number is assigned by the PCI SIG to the end board manufacturer to uniquely identify the manufacturer's name. AMCC does NOT support this feature in the S5335.
- **SID** - The system identification number is assigned by the end product manufacturer under the rights of their SVID assignment. AMCC does NOT support this feature in the S5335.

# Developer's Kit nvRAM Factory Settings

The following are the factory-programmed settings for the nvRAM to run the SRAM design examples:

| Base Address | nvRAM Value | Type |
|---|---|---|
| BADDR0 | 10e8ffc0 | Memory Mapped, 128 bytes, 32-bit |
| BADDR1 | fffe0000 | Memory Mapped, 128K bytes, 32-bit |
| BADDR2 | fffe0000 | Memory Mapped, 128K bytes, 32-bit |
| BADDR3 | fffe0000 | Memory Mapped, 128K bytes, 32-bit |
| BADDR4 | fffe0000 | Memory Mapped, 128K bytes, 32-bit |

| Address | nvRAM Value | Type |
|---|---|---|
| 45 | 81 | PCI side initiated bus mastering Latency Timer PCI 2.1 Compliant |

**IMPORTANT NOTE**: The developer's Kit hardware and software has been designed to operate using the Base Address and Configuration Register values indicated above. Altering these values may cause improper operation. Designers may change settings with the appropriate software and hardware design changes.

# Developer Recommended Documentation

AMCC Data Sheet: S5335 PCI Interface (supplied)
PCI Local Bus Specification, Revision 2.1 (PCI SIG)
PCI BIOS Specification, Revision 2.1 (PCI SIG)
Other related Applications Notes and Design Notes can be downloaded from the AMCC website at:

http://www.amcc.com

To obtain listed documentation from the PCI SIG, contact:

PCI Special Interest Group
P.O. Box 14070
Portland, OR 97214
(800) 433-5177
(503) 797-4207
FAX (503) 234-6762

Chapter 2

# Hardware Installation

## System Requirements

The card and software contained in the Developers Kit are designed to run in a Windows 98/2000 environment. The system requirements are generally the same as listed for the Windows operating system.

The recommended system requirements are:

- 486DX processor or better
- 128 MB system RAM memory
- 10 Meg Hard Disk
- 3.5” Floppy Disk Drive
- CD-ROM Drive
- Windows 98/2000
- 256-color VGA Display
- Keyboard
- PCI bus motherboard slots

## Installing the Hardware

The following section details the installation procedure for hardware components contained in the Developer’s Kit. This developer’s kit is intended and designed for an electronics laboratory environment in which the PC containing the S5335 PCI card will remain open. This allows access to special connectors for logic analyzer. Be sure that all AC power has been removed from your computer before proceeding. AMCC recommends all installation work be done at a static free workstation. If one is not available, ensure that you have removed the static charge from your cloths by touching an object made of metal on the computer or ESD ground on the workstation before proceeding.

### *Installing the PCI Card*

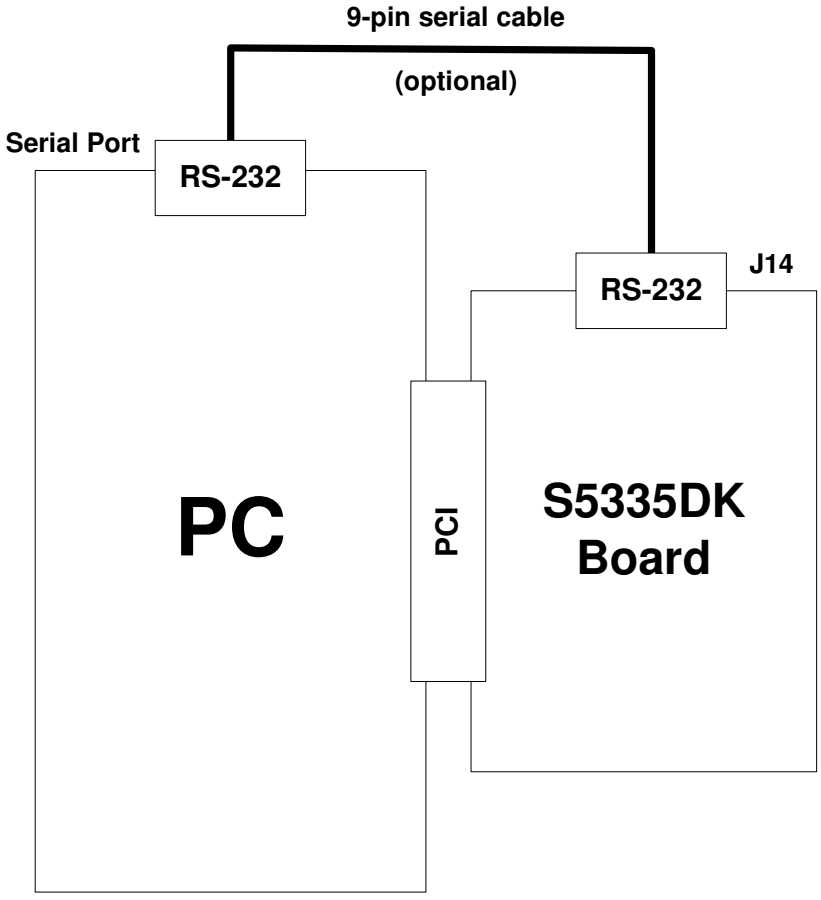
- Remove the cover mounting screws on your computer and carefully remove the cover. Store the cover in a safe place.
- Ensure the S5335 PCI Card’s jumpers are installed per factory setting. See chapter 7 for jumper description to configure for other required options.
  - JP1 – OPEN
  - JP2 – OPEN
  - JP3 – SHORT

- J15 – SHORT pin 1 and 3, 2 and 4
- J18 – SHORT pin 1 and pin 2, 3 and 4

- Hold the Developer PCI Card by its top corners and insert into any available PCI slot. Press down gently but firmly until the card is seated.
- Follow all suggested safety guidelines in your computer manufacturer's manual.

## *Installing the 9-pin RS-232 cable (optional)*

The S5335 PCI card may be used to access the AddOn side to read and write registers and initiate bus mastering operations through the 9-pin RS-232 connector. In order to access the AddOn side from the PC, the 9-pin cable must be installed per figure below in order for the AMCCPCI5335AddOnWin software application to work properly. The cable connects on the back of the computer from S5335DK J14 to the PC's serial port. The cable is a one-to-one cable with a female 9-pin connector on one end and a male 9-pin connector on the other end.



# Connecting the Hewlett-Packard Logic Analyzer

The S5335 PCI card was developed with a set of conveniently located logic analyzer connectors. These connectors were designed to connect to a custom proto-type board or a logic analyzer. The proto-type board connects to the S5335 Add-On bus through connectors J1, J2, J5, and J6 on the component side of the S5335 PCI card. The logic analyzer connects to the Add-On bus through the same connectors. When connected to the logic analyzer, it allows the developer to examine the Add-On bus timing. The four connectors contain all the signals of the Add-On bus and ground references. These connectors are pin designated for direct pod cable connection to the Hewlett Packard 16500B logic analyzer. Up to four cables may be connected to cover the entire Add-On bus signal set. Refer to the schematics for signal location before connecting the HP or any other logic analyzer.

Chapter 3

# Software Installation

## Installing the Software

Installing the software on a Windows system is a two-step process. The first step is to install the S5335 PCI driver for the card, which involves booting the system after the card is installed and following the normal Windows procedure for recognizing new hardware in the system. The second step is to install the Windows AMCCPCI5335Win and AMCCPCI5335AddOnWin applications that will allow access to the features on the card.

### S5335 Driver Installation

- Turn power off to the PC and install the S5335DK board
- Turn power on to the PC and boot Windows

**Windows 2000:**

- Windows will detect the new hardware and start the “Found New Hardware Wizard”.
- Put the CDROM for the S5335DK board in the CDROM drive
- At the first Wizard screen, “Welcome to the Found New Hardware Wizard”, and click on Next.
- At the “Install Hardware Device Drivers” screen, select the “Search for a suitable driver for my device (recommended)” option, and click on Next
- At the “Locate Driver Files” screen, select “Specify a location”, and click on Next
- At the “Insert the Manufacturer’s installation disk in the drive selected, and click OK” screen, click on the “Browse” button.
- From the “Browse screen, navigate to the CDROM directory: “S5335 Software\ Driver Installation” and then click on Open. Control goes back to the previous screen.
- Back at the “Insert the Manufacturer’s installation disk in the drive selected, and click OK” screen, click on the “OK” button.
- At the “Driver Files Search Results” screen, the message should be displayed indicating that “Windows found a driver for this device, and display the name of the INF file found for it: “S5335 Software\ Driver Installation \siwdm_5335.inf” and then click on Next.
- Finally, the last screen “Completed the Found New Hardware Wizard” is displayed. Click on “Finish”.

**Windows 98:**

- Windows will detect the new hardware and display a “New Hardware Found” message.
- Put the CD-ROM for the S5335DK board in the CD-ROM drive
- Windows will start the “Add New Hardware Wizard”. From the first screen click on Next.
- At the “What do you want to do now?” screen, click on the “Search for the best driver for your device (recommended)” option, and click on Next
- At the “Windows will search…” screen, select “Specify a location” and click on the Browse button. Navigate to the following directory “S5335 Software\ Driver Installation” and click on OK. This goes back to the “Windows will search…” screen. Click on Next.
- At the “Windows driver file search for the device” screen, it should say “Location Found”. Click on Finish to complete the installation.

- For either Windows system, the driver is now installed.

## AMCCPCI5335Win Application Installation

- Put the CD-ROM for the S5335DK board in the CD-ROM drive
- Navigate to the following CD-ROM directory: “S5335 Software\ Application Installation\ PCI Side” and double-click on SETUP.EXE. This starts the application installation process.
- Follow the instructions to install the application on the PC. It is generally acceptable to select all default values for each screen and then click on Next to go to the next screen.
- Once installed, an icon is placed on the Desktop. This icon can be used to start the application.

## AMCCPCI5335AddOnWin Application Installation

- Put the CD-ROM for the S5335DK board in the CD-ROM drive
- Navigate to the following CD-ROM directory: “S5335 Software\ Application Installation\ AddOn Side” and double-click on SETUP.EXE. This starts the application installation process.
- Follow the instructions to install the application on the PC. It is generally acceptable to select all default values for each screen and then click on Next to go to the next screen.
- Once installed, an icon is placed on the Desktop. This icon can be used to start the application.

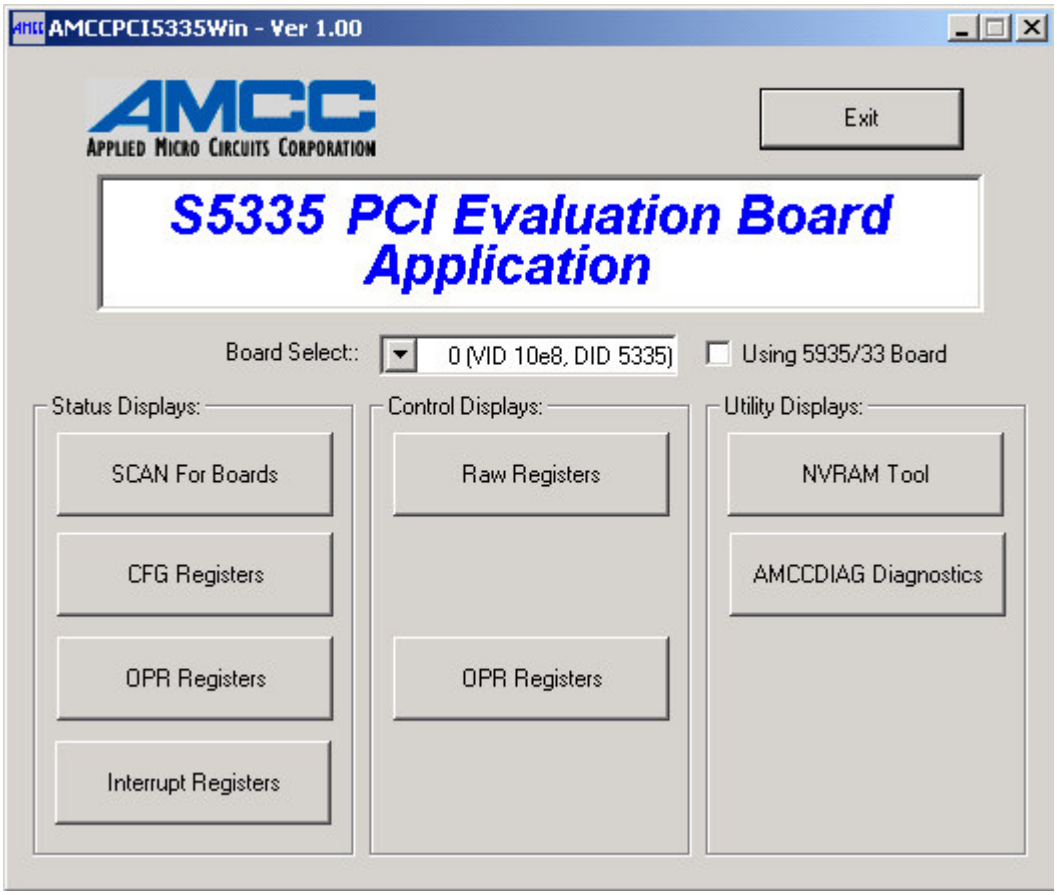
# System Checkout

After installing the hardware and software as described in the previous sections, an operational test should be run to ensure proper Developer's Kit system functioning. This will ensure the motherboard, system BIOS, PCIBIOS, Windows, the S5335 Application and the Developer's Kit hardware are all working properly and are in sync. Certain aspects have already been tested, such as the recognition of the hardware and the driver installation. The following steps will operate various aspects of the S5335DK to verify the remaining operation.

## *PCI Side Checkout (AMCCPCI5335Win Application):*

**STEP 1:** Test the presence of the S5335DK.

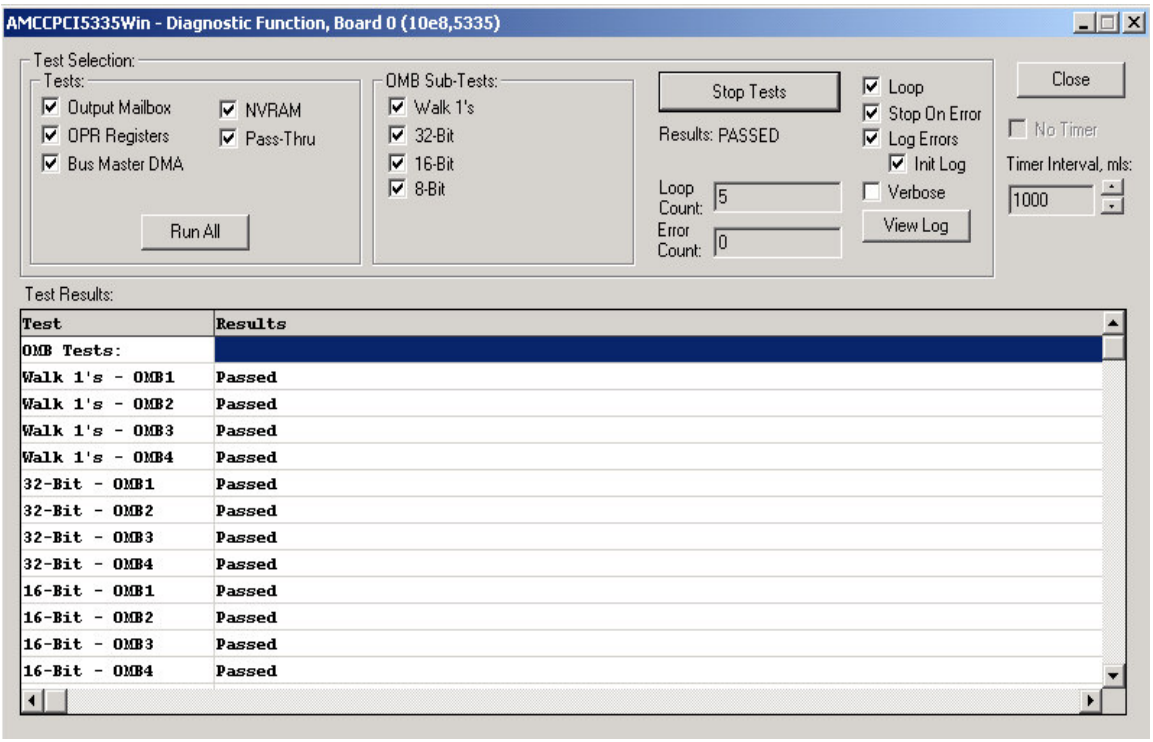
- Run the AMCCPCI5335Win.EXE program from Windows.
- First note that the front screen comes up similar to the image below, and that a message indicating "No Boards!" is NOT displayed. Using the "Board Select" list box on the front screen, open the list box by clicking on the arrow and verify that there is a board listed that shows an entry with VID=10E8 and DID=5335. Click on this entry so that the list box closes and displays this entry. This indicates a PCIBIOS is present and has located and recognized the S5335 Developer Kit main card. If a board of this type is NOT displayed in the "Board Select" list box, verify the settings in the system BIOS menus and the correct installation of the DK main card.
- Note, if more than one PCI board is present on the system, the Board Select window may have to be clicked to show all boards and allow the correct board to be selected. Click on the arrow in the left side of the "Board Select" window to show all boards in the system, then click on the desired board.




**STEP 2:** Test the S5335DK board

- Run AMCCPCI5335Win.EXE program. With the "Board Select" list box showing the S5335 board selected in Step 1, select the Diagnostics window by clicking on the "AMCCDIAG Diagnostics" button on the front screen.

- The Diagnostics window comes up similar to image below with all tests selected, and with Loop and Stop On Error selected. Click on "Start Tests" to run and display the results of the tests. The results are displayed in the grid area at the bottom of the screen. Note that the message "Results: PASSED" is displayed just below the "Start Tests" (which now says "Stop Tests") button. If "Results: FAILED" is displayed, then errors were detected in the tests and the board is not functioning properly.




- As long as all the tests pass successfully, the set of tests will loop over and over again. Let the tests run until the “Loop Count” shows at least 5. This indicates that at least 5 passes of the tests have completed and all tests have passed successfully each time. When enough passes of the tests have completed, click on “Stop Tests” to stop the running of the tests and click on “Close” to close this screen.

- If there are errors, they will appear in the “Test Results” area of the display. However, this area is also used to display the Passed tests and the error messages may difficult to find. With the “Log Errors” and the “Init Log” boxes checked, the errors will be displayed in the log file on disk. Click on the “View Log” button to display the log file using Notepad to view any errors after the tests have stopped.

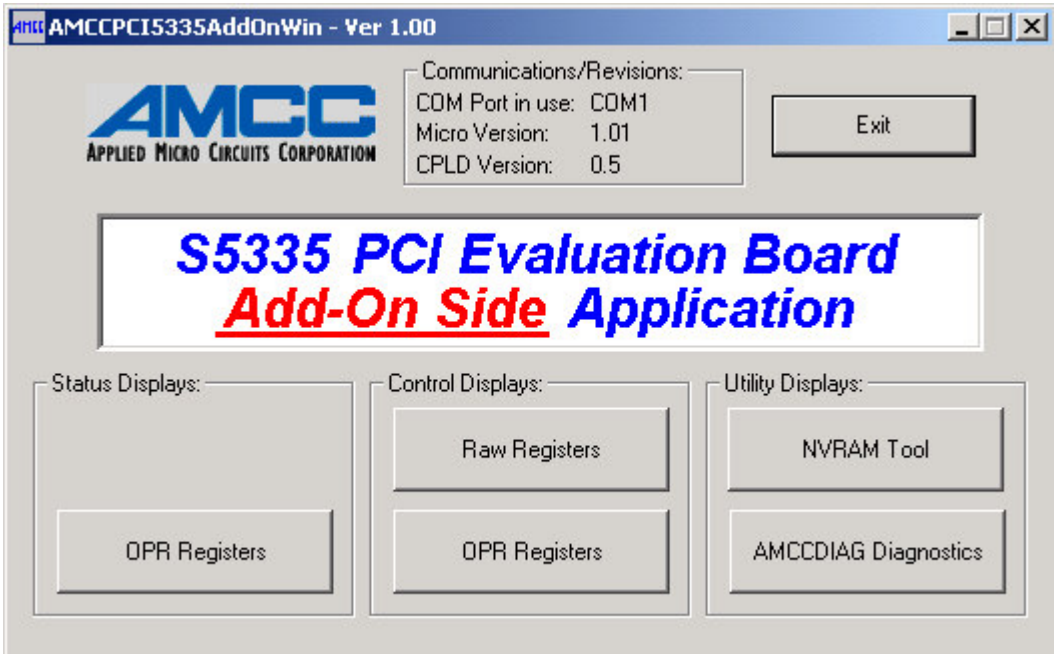
Note: Running the PCI Side Bus Master DMA tests requires that the AMCCPCI5335AddOnWin application is available to setup the AddOn side for proper execution of the DMA tests from the PCI side. The user will be prompted to make sure the setup is correct when the Bus Master DMA diagnostic is first run.

**STEP 3:** The operational tests are complete.

## *Add-On Side Checkout (AMCCPCI5335AddOnWin Application)::*

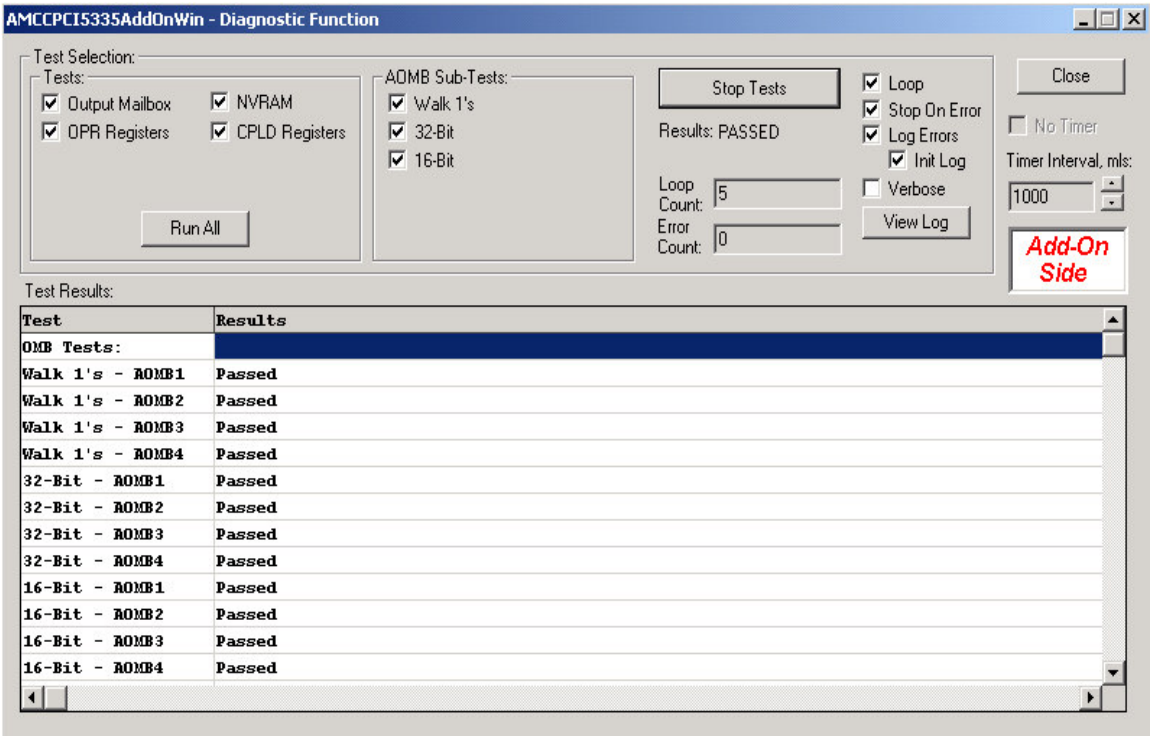
**STEP 1:** Test the presence of the S5335DK.

- Run the AMCCPCI5335AddOnWin.EXE program from Windows.
- First note that the front screen comes up similar to the image below, and that a warning message indicating "No Communications established with 5335…" is NOT displayed.




**STEP 2:** Test the AddOn side of the S5335DK board

- Run AMCCPCI5335AddOnWin.EXE program. Select the Diagnostics window by clicking on the "AMCCDIAG Diagnostics" button on the front screen.
- The Diagnostics window comes up similar to image below with all tests selected, and with Loop and Stop On Error selected. Click on "Start Tests" to run and display the results of the tests. The results are displayed in the grid area at the bottom of the screen. Note that the message "Results: PASSED" is displayed just below the "Start Tests" (which now says "Stop Tests") button. If "Results: FAILED" is displayed, then errors were detected in the tests and the board is not functioning properly.




- As long as all the tests pass successfully, the set of tests will loop over and over again. Let the tests run until the “Loop Count” shows at least 5. This indicates that at least 5 passes of the tests have completed and all tests have passed successfully each time. When enough passes of the tests have completed, click on “Stop Tests” to stop the running of the tests and click on “Close” to close this screen.

- If there are errors, they will appear in the “Test Results” area of the display. However, this area is also used to display the Passed tests and the error messages may difficult to find. With the “Log Errors” and the “Init Log” boxes checked, the errors will be displayed in the log file on disk. Click on the “View Log” button to display the log file using Notepad to view any errors after the tests have stopped.

Note: Running the AMCCPCI5335AddOnWin AddOn side tests does NOT require the that the AMCCPCI5335Win application is running.

**STEP 3:** The operational tests are complete.

# Chapter 4
# AMCCPCI5335Win Software Application

The AMCCPCI5335Win application is a Windows 98/2000 application that can do the following:

- Display Configuration and Operational registers
- Allow the Operational Registers to be modified
- Modify and Program the nvRAM
- Run Diagnostics

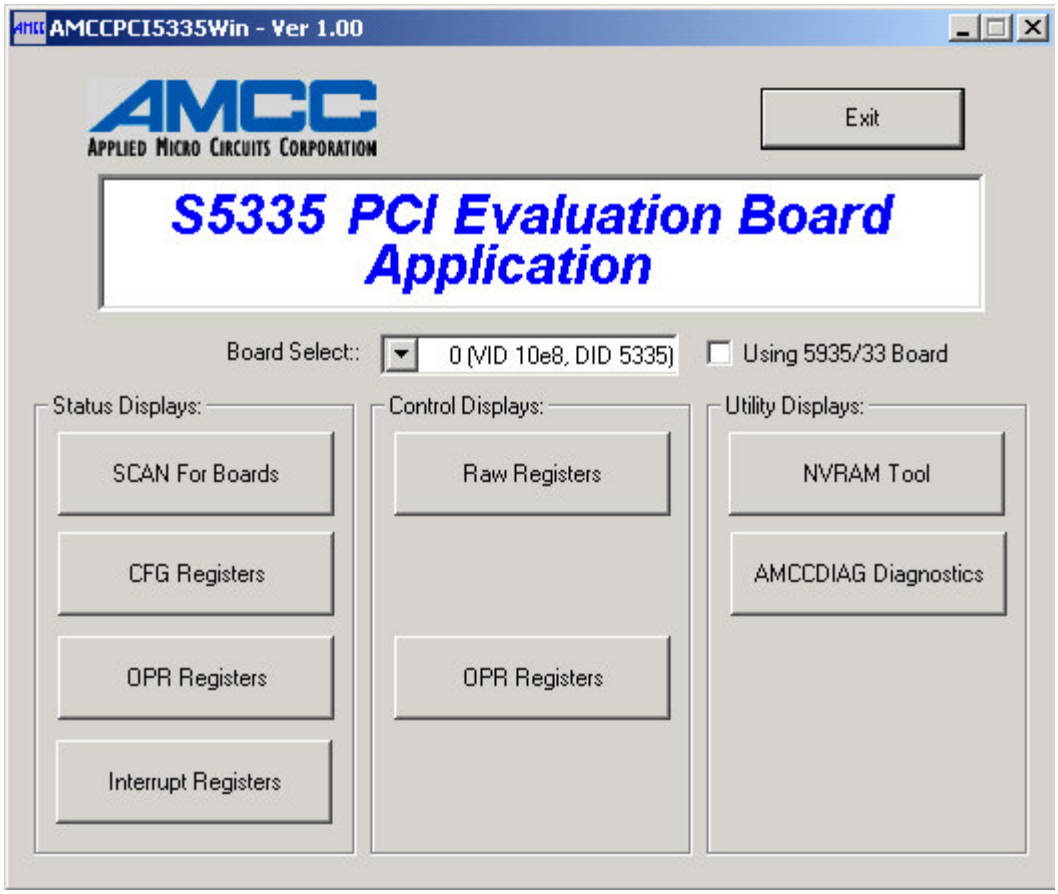
Besides the Diagnostics used for System Checkout discussed previously, there are a number of other screens and features contained in the AMCCPCI5335Win application. These are explained in the following section.

Multiple S5335DK boards (up to four) in the same PC may be accessed by this application by merely selecting the board and then selecting the function desired for that board. The same function may be run on different boards by bringing up the function for one board, then selecting a different board and bring up that same function again.

Note that source code for this application is NOT supplied on CD. Source code is supplied for a smaller version of this application that contains only the NVRAM Utility portion of this application. This smaller application is AMCCNVRAMWIN. The source code contains complete code for the application, library and driver to support the nvRAM access only.

## Main Screen

This screen is the first screen displayed after the AMCCPCI5335Win application is started.

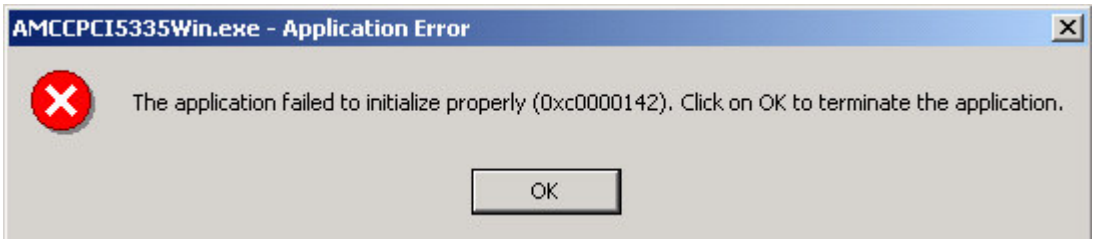



This screen is divided into 4 areas. The first area is the “Board Select” window. This is used to select the S5335DK board on which to operate. The “Status Displays” will display static information from the registers. The “Control Displays” will allow the settings in the registers to be changed. Finally, the “Utility Displays” will allow other settings and values to be changed on the board.

If multiple PCI cards are present in the system, they will all be displayed in the “Board Select” list box. Before bringing up any other screens in this application be sure to select the proper S5335 Development board in this list. Each additional screen uses the board selected from this list when it is brought up.

Note that if no PCI cards are detected, the following error messages may be displayed and the AMCCPCI5335Win application will not run properly:


and then (perhaps) something like the following:

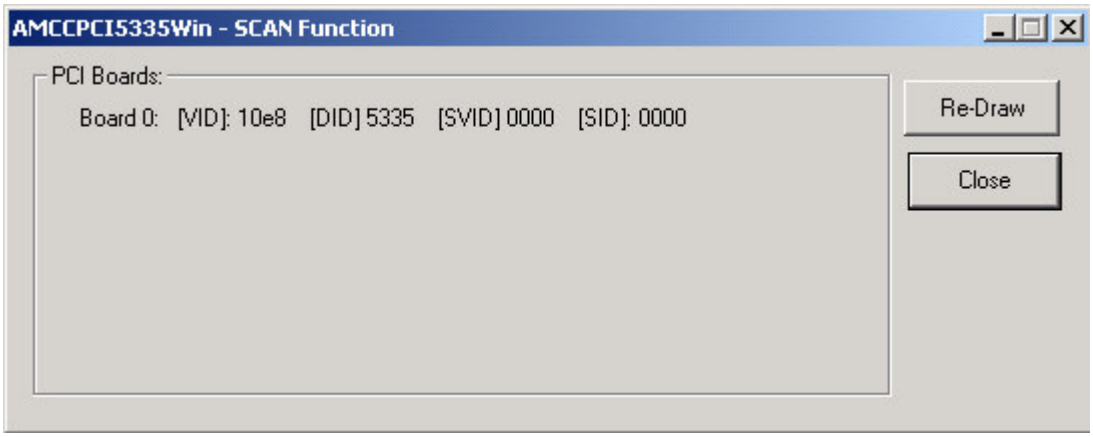



If these messages are displayed, the condition causing the failure of recognition of any boards must be corrected and the application restarted.

## Status Displays

### SCAN For Boards:

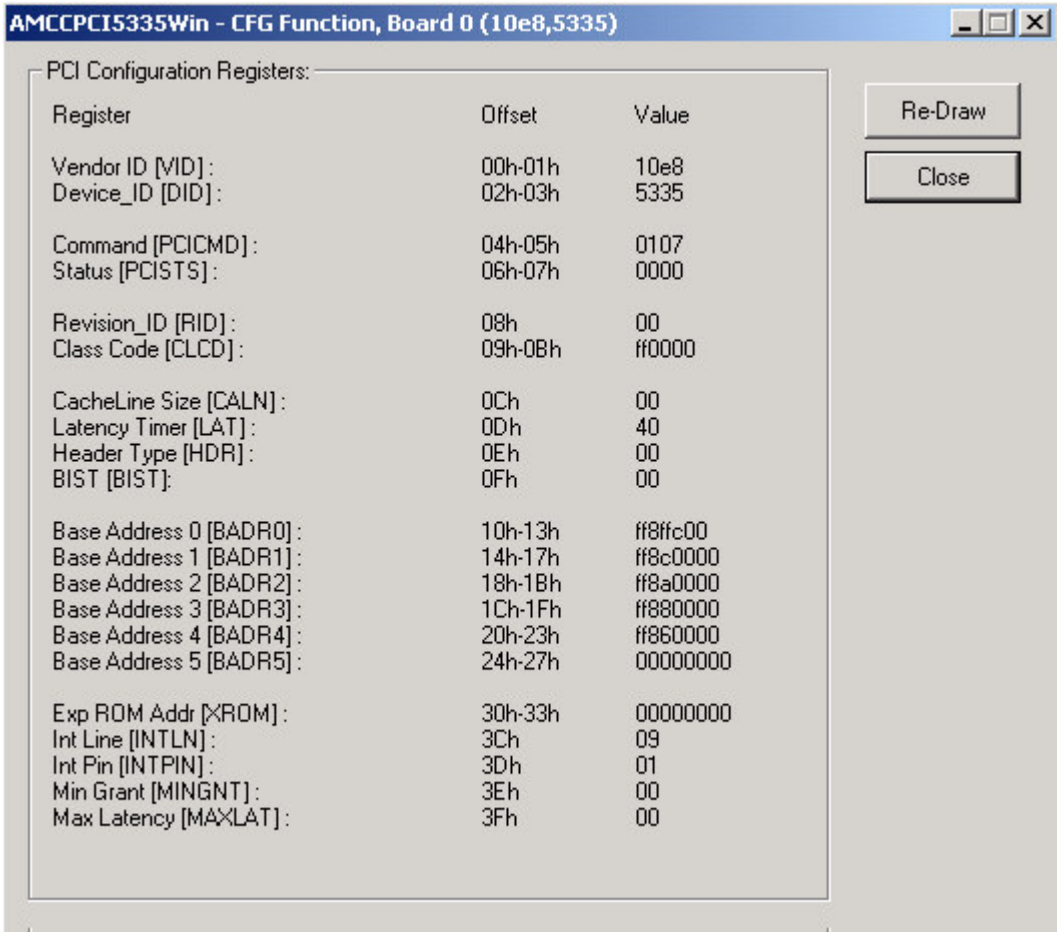
This static display shows the PCI cards currently detected in the system by the AMCCPCI5335Win application and driver. These boards may not all be S5335 boards. Clicking on the “Re-Draw” button will re-scan for PCI boards and display the results again.



**CFG Registers:**

This static display shows the Configuration Register values for the board selected. The board is selected from the front screen before this screen is displayed. Note that the screen's menu bar displays the board number. Clicking on the "Re-Draw" button will re-read the Configuration Registers for the board and display the results again.

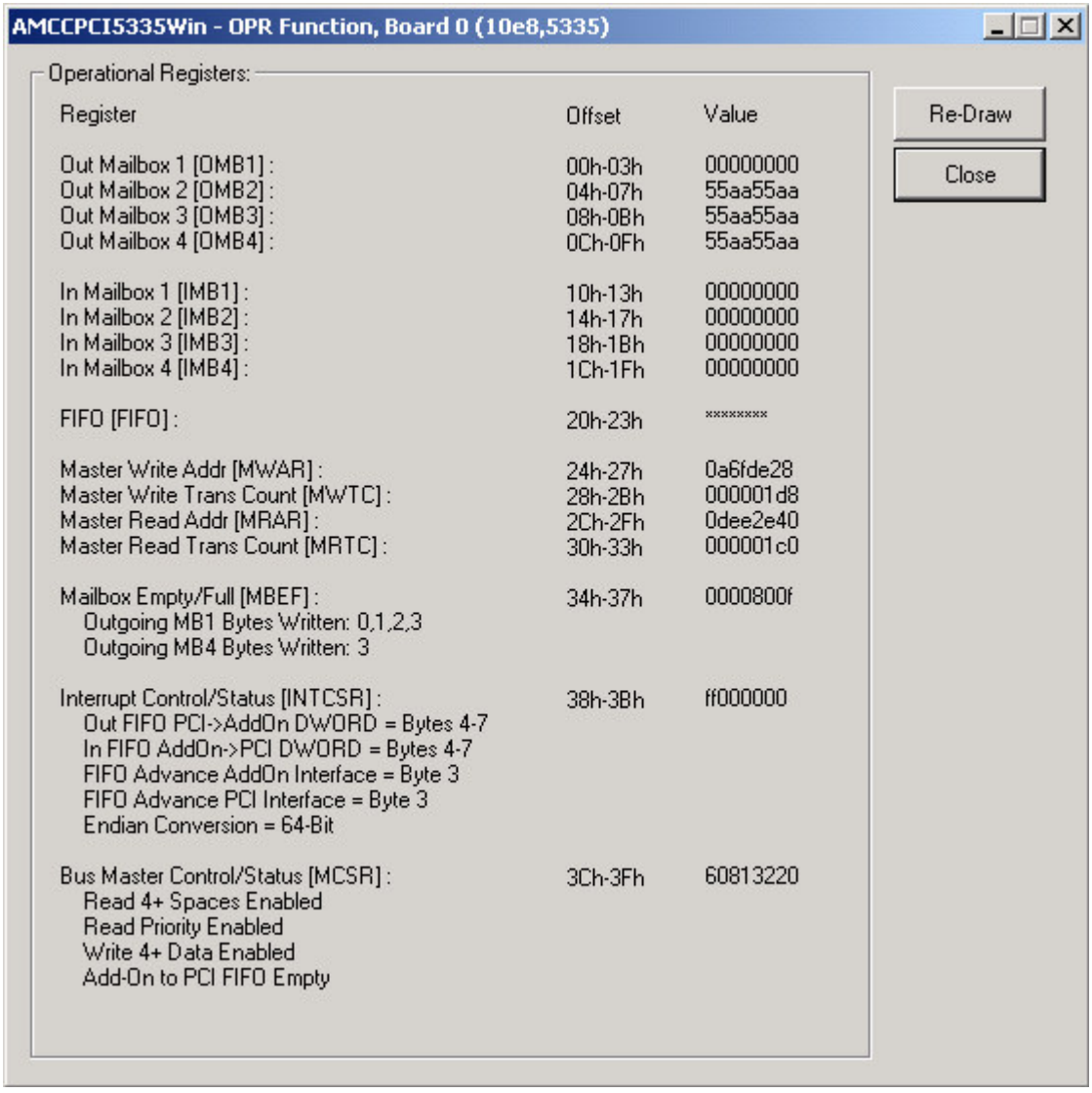
The image below shows a typical Configuration Register display. It shows the register name, the Offset address and the current Value for the registers.



**OPR Registers:**

This static display shows the Operational Register values for the board selected. The board is selected from the front screen before this screen is displayed. Note that the screen's menu bar displays the board number. Clicking on the "Re-Draw" button will re-read the Operational Registers for the board and display the results again.

The image below shows a typical Operational Register display. It shows the Register name, the Offset address and the current Value for the registers. In addition, it shows the state of some of the bits within the registers.

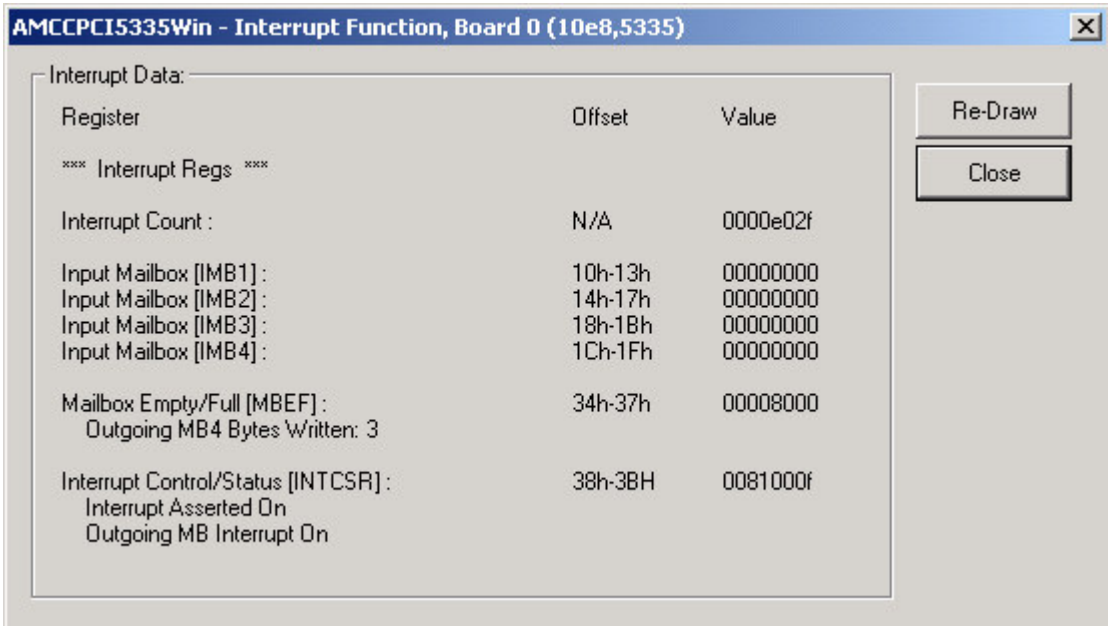



Note that this is a static display and no Operational Register values can be changed from this screen. See the OPR Control screen description for changing Operational Register values.

**Interrupt Registers:**

This static display shows various Operational registers trapped during an interrupt operation, most notably during a Bus Master DMA operation. In addition, certain interrupt counts are displayed. The board is selected from the front screen before this screen is displayed. Note that the screen's menu bar displays the board number. Clicking on the "Re-Draw" button will re-read the Configuration Registers for the board and display the results again.

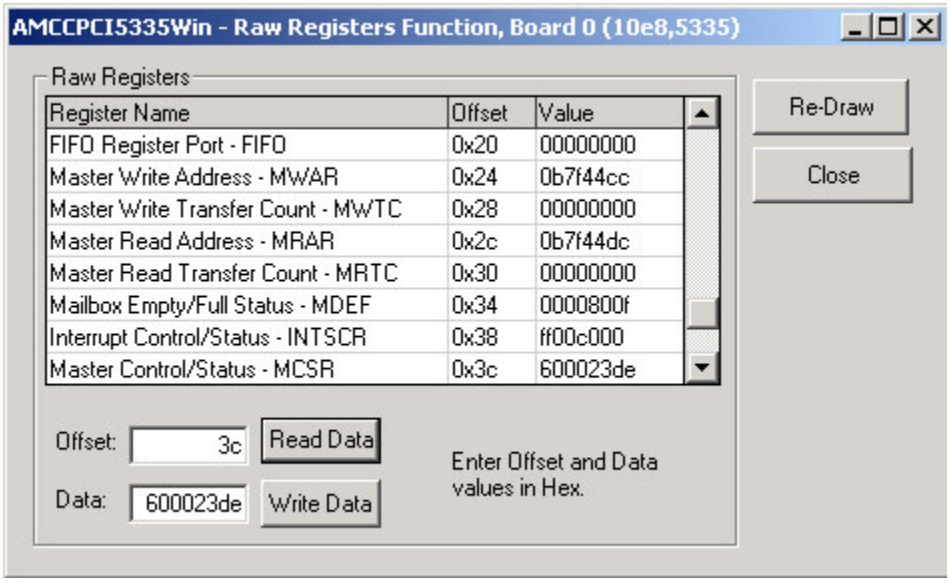
The image below shows a typical Interrupt Register display. It shows the register name, the Offset address and the current Value for the registers.



## Control Displays

### Raw Registers:

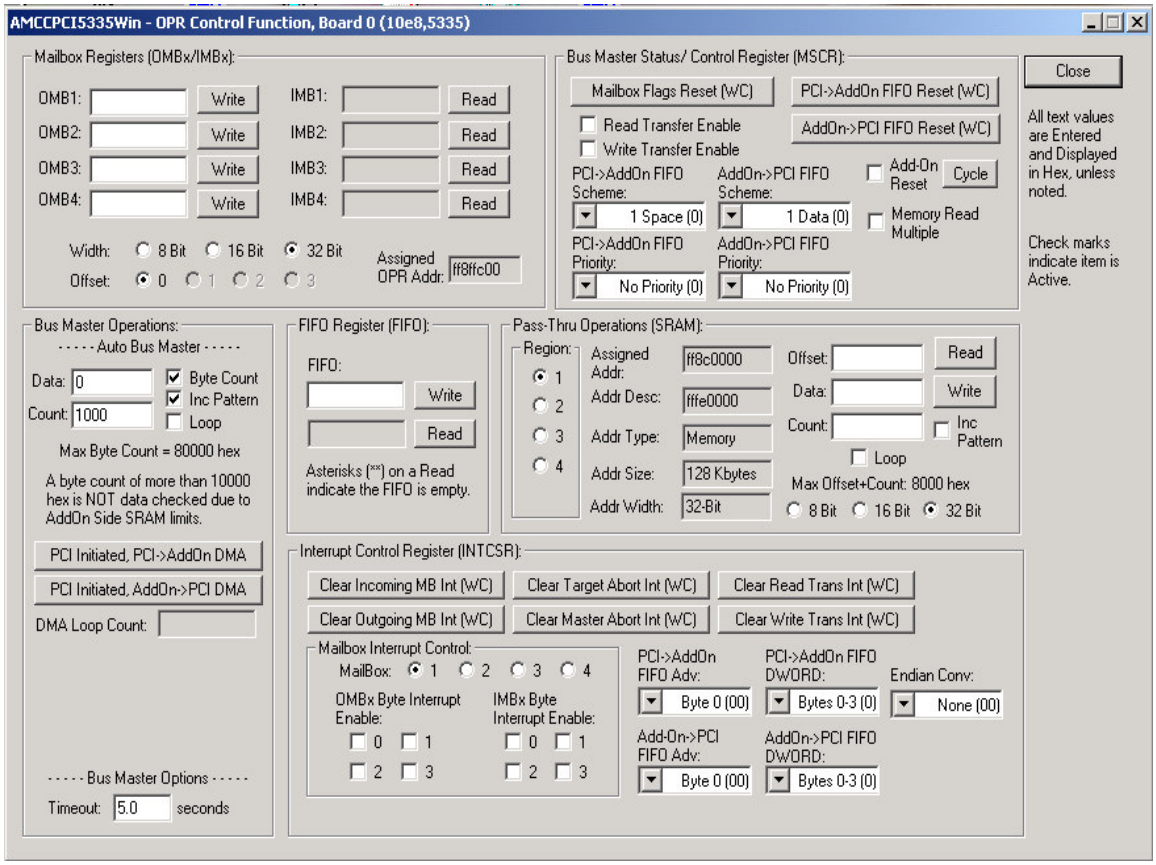
This screen allows the programming of the Operational Registers by reading and writing full 32-bit and/or 8-bit (CPLD) registers.




This screen consists of a list of the registers showing the values for the registers in the grid area. By entering the “Offset” and clicking the “Read Data: button, the register at the offset will be read and displayed in the “Data” box, as well as in the grid. In addition, by entering the “Offset” and “Data” values in the respective boxes, and clicking the “Write Data” button, the data entered will be written to the register.

**OPR Registers:**

This screen allows the programming of the Operational Registers to the settings required for a particular test or usage of the board by programming the individual functional fields within the registers.




This screen consists of many buttons and controls that easily allow the setting of the Operational Register controls. There are basically four types of controls on this screen: buttons, check boxes, data entry boxes and list boxes containing a list of valid selections.

When clicked, the buttons cause an action to take place. This action may be the writing of a single bit (“PCI->AddOn FIFO Reset (WC)”, for example), the reading of a text entry box and writing this value to a register (“Write” button in the SRAM controls), or a sequence of steps to carry out an action (“Cycle Add-On Reset”).

The check boxes allow the setting or clearing of a single bit control. When clicked and the check box has a check mark, the indicated control is on, or active. When clicked and the check box does not have a check mark, the control is off, or inactive.

The data entry boxes allow a value to be typed and when an associated “Write” or “Read” button is clicked, the information contained in this box is used for the

operation. No action is taken to use any of the data entered until the associated button is clicked to start the action. Note that there are some boxes that white and others that are gray, or darkened. Only in the white boxes can data be entered. The gray boxes are read only and are used to display values only.

The list boxes operate very much like the data entry boxes and their associated button. The difference is that for the controls that use list boxes, there are only a small number of correct entries and we list them all in the list box. When the arrow in the box is clicked, all the items in the list are displayed (scrolling may be required to see them all), and the desired entry should be clicked. The list box will then be closed with the item selected displayed, and the value that corresponds to the item selected will be written to the register.

Note that anytime a change is made using this control screen, an update command is sent to the OPR Registers status display, if it is displayed. Any changes, therefore, to the OPR registers should be automatically displayed in the status window.

**Mailbox Bytes (OMBx/IMBx):**

These controls allow the writing of the four OMB registers and the reading of the four IMB registers. First select the bit width of the operation, 8-bit, 16-bit or 32-bit, and select the offset associated with the bit width of the operation. Note that the available offsets will change as the bit width is changed. For a write to the OMB register, enter the data to be written. Then click the “Write” button to write to the OMB register, or the “Read” button to read from the IMB register. For a read operation, the data read will be displayed in the “IMB:” box (grayed out).

Note that there is a grayed out box labeled “Assigned OPR Addr:” This box displays the address being used, the one assigned by the system, to access the Operational Registers. This is for informational purposes only.

**Interrupt Control Register (INTSCR):**

These controls allow the writing of the individual fields of the Interrupt Control Register, or INTSCR. The six buttons, such as “Clear Incoming MB Int (WC)” and “Clear Outgoing MB Int (WC)” will write a one to the corresponding interrupt bit.

The two groups of four check boxes for OMB Byte Interrupt Enable and IMB Byte Interrupt Enable, will enable (checked) or disable (not checked) the interrupt enable by building the command and issuing it to the Interrupt Control Register.

Finally, the five list boxes that are used to set the FIFO advance, and Endian controls for manual Bus Master DMA operations.

**Master Control/Status Register (MCSR):**

These controls allow the writing of the individual fields of the Master Control /Status Register, or MCSR. The three buttons, “PCI->AddOn FIFO Reset (WC)”, “AddOn->PCI FIFO Reset (WC)” and “Mailbox Flags Reset (WC)” will write a one to the corresponding reset bit.

The “Add-On Reset” check box enables (checked) or disables (not checked) the Add-On Reset bit. The “Cycle Add-On Rest” button will automatically cycle the Add-On Reset bit by enabling it, then disabling it.

The “Memory Read Multiple” check box will set (checked) or clear (unchecked) the bit in the register.

In addition there are four list boxes to allow the selection of the desired FIFO scheme and priority when using manual Bus Master DMA operations.

**Bus Master Operations:**

These controls allow the user to initiate Bus Master DMA operations to or from the AddOn side.

The data pattern and the byte count should first be entered in the “Data” and “Count” boxes. The Data will be used to generate data pattern for the DMA operation. The “Byte Count” check box indicates that the count entered is a byte count. If this box is unchecked, the count will be a 32-bit DWORD count and the count entered will be multiplied by 4 to get the byte count. Note that the Data and Count values are always entered in Hexadecimal.

Note the maximum byte count and data checking byte count information that is displayed. While these maximums are not the maximum that the S5335 chips can DMA, they are the maximum that can currently be done by this Windows application at this time.

If an incremental pattern is selected by checking the “Inc Pattern” box, then the pattern generated will be incremented from one value to the next. If not selected, then the same pattern will be generated from one value to the next.

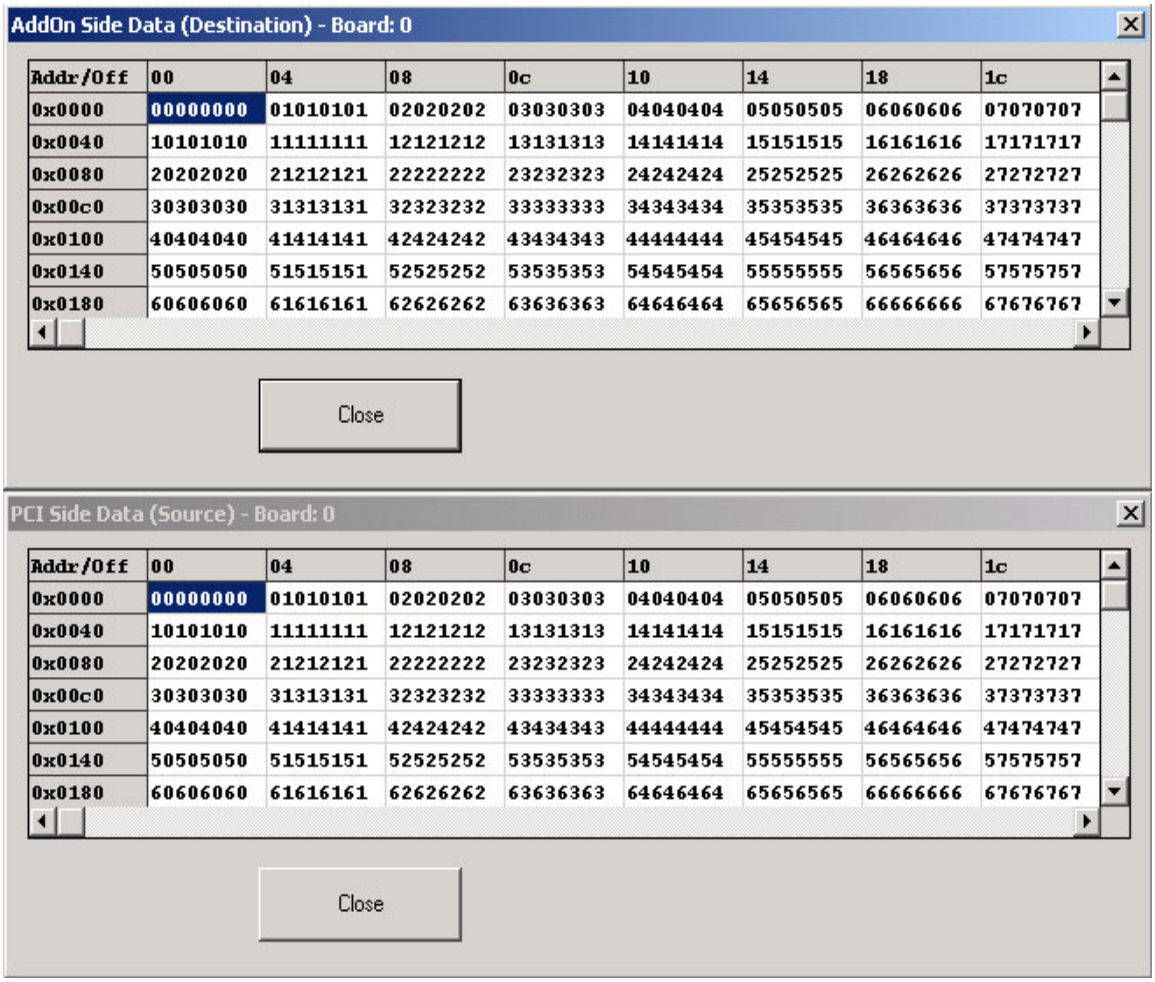
If the operation is to be looped, then check the “Loop” check box. When the operation is looped, the same operation will be looped over and over again until the loop check box is unchecked.

The “Timeout:” box can be used to change the length of time to wait for the DMA operation to complete. The time out can be entered to a tenth of a second.

In addition, FIFO advance and Endian options for the DMA operation can be set from the INTSCR selections for these values.

After all options are selected, click on either the “PCI Initiated, PCI->AddOn DMA” or “PCI Initiated, AddOn-> PCI DMA” buttons to do the operation. All DMA operations involve either reading or writing data to the SRAM contained on the board. Depending on the operation, either the SRAM will be written with data before the DMA operation is initiated, or the SRAM will be read after the DMA operation is completed.

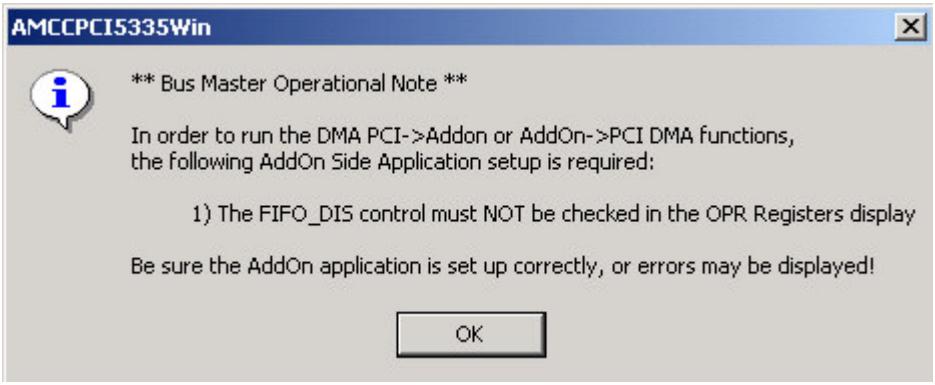
Depending on the options selected, after the operation is complete both the source and destination data will be checked and displayed. Any errors in data will be indicated, and the data buffers displayed. The data displayed may be only the first part of the buffers if the byte count is larger than the maximum display size. The data is displayed in a format similar to the following, showing both the source and destination data:




The “DMA Loop Count:” box will display the number of DMA operations completed, but are only useful when the operation is looped.

**Note: In order to initiate a Bus Master DMA operation, the “FIFO_DIS” control on the AddOn side must be unchecked. The user will be**

**reminded of this each time the first DMA operation is attempted with the following message:**




**Note: If the two "PCI Initiated,…" buttons are grayed out, then the NVRAM indicates that Bus Mastering is enabled from the AddOn side, not from the PCI side. In this case, two additional buttons will be displayed allowing the user to "Connect To AddOn Side…" DMA operations. The use of these two additional buttons is explained in the AddOn side application information.**

**FIFO Register (FIFO):**
These controls allow the reading and writing of the FIFO register.

To write to the FIFO, enter the 32-bit data value in the box, in hex, and click on the "Write" button.

To read from the FIFO, click on the "Read" button. The data read will be displayed in the gray box. Note that if the FIFO is empty, the FIFO is NOT read and asterisks will be displayed. This is a driver requirement.

**Pass-Thru Operations (SRAM):**
These controls allow the setting of parameters to control the Pass-Thru Configuration Register, and also carry out reads and writes to the SRAM on the S5335Dk board that is used to test the Pass-Thru bus.

First, select the Region to be working with. This is done by selecting one of the four regions by clicking on one of the select controls (circular radio buttons) 1 thru 4. When this is done, the rest of the controls in the Pass-Thru area are set to the current settings for that region and all changes will be directed to that region.

As the region is selected, there are a number of grayed out (read only) text boxes that will display information about how the region is accessed. This information includes the Assigned Address, and if the nvRAM exists, the

Address Descriptor, the Address Type, the Address Size and the Address Width. All of this information is displayed for informational purposes only.

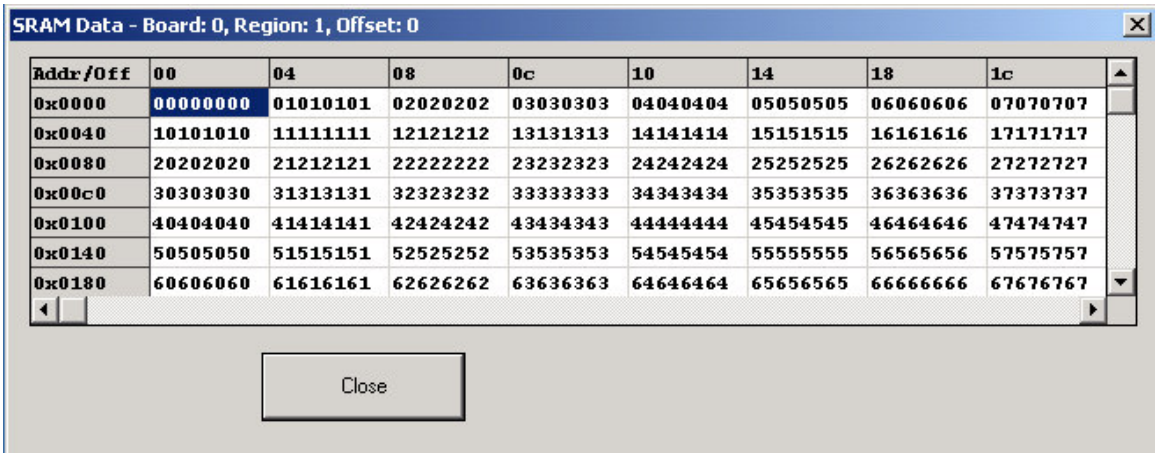
The remainder of the controls is intended for using the SRAM interface using region 1. These controls allow the setting of the starting Offset and the Count and bit width for the SRAM accesses. These two items of information must be entered. Note that the Offset is always the byte offset to start the operation and the Count is the count of the number of data items (not bytes) for the operation. The maximum sum of the Offset and Count are always displayed, and will vary as the bit width is changed.

In addition, if a write operation is desired, the Data for the write should be entered, and an incremental pattern indicator ("Inc Pattern" check box) should be selected if desired, before the write operation is started. If a read operation is desired, the Data does not have to be filled in, and the Data box may be written with the data read.

If an incremental pattern is selected, then the pattern generated will be incremented from one value to the next. If not selected, then the same pattern will be generated from one value to the next.

Note that the Offset, Data and Count values are always entered in Hexadecimal.

Finally either the "Read" or "Write" button should be clicked to do the operation. If "Write" is clicked, then the write operation is done, which means the write data is first generated and then written to the SRAM. If "Read" is clicked, the data is read from the SRAM and displayed. The display of the data will be one of two ways. If a count of one was entered, the data read will be displayed in the Data box under the SRAM controls. If the count is more than one, then a separate window will be generated showing all of the data read. The window will look similar to the following screen.



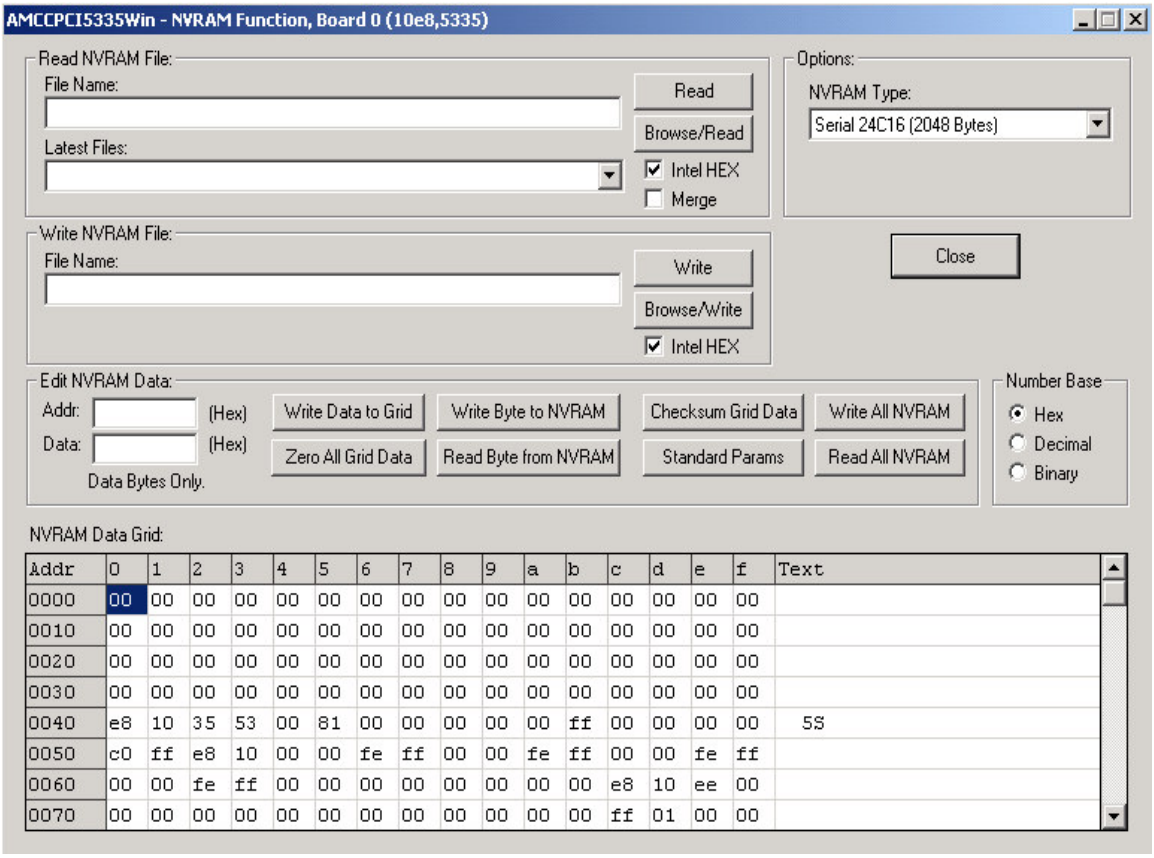

| Addr/Off | 00 | 04 | 08 | 0c | 10 | 14 | 18 | 1c |
|---|---|---|---|---|---|---|---|---|
| 0x0000 | 00000000 | 01010101 | 02020202 | 03030303 | 04040404 | 05050505 | 06060606 | 07070707 |
| 0x0040 | 10101010 | 11111111 | 12121212 | 13131313 | 14141414 | 15151515 | 16161616 | 17171717 |
| 0x0080 | 20202020 | 21212121 | 22222222 | 23232323 | 24242424 | 25252525 | 26262626 | 27272727 |
| 0x00c0 | 30303030 | 31313131 | 32323232 | 33333333 | 34343434 | 35353535 | 36363636 | 37373737 |
| 0x0100 | 40404040 | 41414141 | 42424242 | 43434343 | 44444444 | 45454545 | 46464646 | 47474747 |
| 0x0140 | 50505050 | 51515151 | 52525252 | 53535353 | 54545454 | 55555555 | 56565656 | 57575757 |
| 0x0180 | 60606060 | 61616161 | 62626262 | 63636363 | 64646464 | 65656565 | 66666666 | 67676767 |

One last note is about the Loop check box. This check box is there only for debugging purposes. If checked, the Read or Write operation will be looped continuously. This gives the ability to perhaps look at some of the signals with a scope or logic analyzer while repeatedly executing the operation. To end the loop, merely un-check the Loop box. Note that if a Read operation is done with the Loop checked the data is NOT displayed.

## Utility Displays

### NVRAM Tool:

This screen allows the building of an nvRAM image and/or the reading and writing of an image from/to the nvRAM.




When the screen is first brought up, the existing nvRAM data from the board is read and displayed in the Data Grid area. So this will first show the existing nvRAM data. After this, a disk file may be read with new data, the data may be modified, written back to the nvRAM or a disk file written with the data.

#### Read NVRAM File:

This section allows the reading of a disk file into the Data Grid. First, set the parameters for the file by indicating if the file is an Intel Hex format file (check the “Intel HEX” box) or is a binary file (uncheck the “Intel HEX” box). An Intel Hex file has a specific ASCII format to it, while a binary file is treated as bytes of data, one byte in the file is one byte of nvRAM data.

Next, indicate whether the data in the file is to be overlaid on the existing data in the Data Grid (check the “Merge” box), or should the data in the Data Grid be cleared (uncheck the “Merge” box) before the data is loaded.

Next, the file needs to be selected. The name of a file to read can be typed into the “File Name” box and then read by clicking on the “Read” button, but most often the “Browse/Read” button is going to be clicked which will bring up a file browser window. This browser window can then be used to navigate to the desired file. From the browser box, the file can be doubled clicked to load it into the Data Grid, or can be highlighted (single clicked) and the “Open” button clicked to load the file.

At this point, the file is opened, the data is read and the data is displayed in the Data Grid area.

Note that the file name read is now displayed in the “File Name” box. This box always displays the last file read. If this file is to be read again, just click on the “Read” button and the file will automatically be read without having to browse again.

Additionally, as each file is read, the name is placed in the “Latest Files” box. This keeps a list of all files read. If a file needs to be re-read, and it is not the last file read, pick the file from this list and it will be loaded automatically without browsing. Just make sure the “Intel HEX” and “Merge” boxes are set correctly.

**Write NVRAM File:**

Writing a file to disk is very similar to reading a file, but in this case the data from the Data Grid is written to a disk file. . First, set the parameter for the file by indicating if the file is to be an Intel Hex format file (check the “Intel HEX” box) or is to be a binary file (uncheck the “Intel HEX” box).

Next, the file needs to be selected. Again, the name of a file to write can be typed into the “File Name” box and then written by clicking on the “Write” button, but most often the “Browse/Write” button is going to be clicked which will bring up a file browser window. This browser window can then be used to navigate to the desired file or directory. From the browser box, the file can be doubled clicked to write the data from the Data Grid, or can be highlighted (single clicked) and the “Open” button clicked to write the file. In either of these cases, the existing file will be over written and the original contents will be lost.

If the file is a new file, then navigate to the directory for the file, type the name of the file in the box provided and then click on “Open”. The new file will be opened and written with the data.

**Edit NVRAM Data:**

This is used to modify the data in the Data Grid or read and write data to the nvRAM. The data bytes in the Data Grid can only be edited using the “Addr” and “Data” boxes. Type the address of the data in the “Addr” box and place the new data value in the “Data” box. Note that the number base required for

each entry is displayed just to the right of each box in parentheses. Once the address and data have been entered, the other buttons can be used to do something with them.

Rather than typing the address and data information into the boxes, an alternate way to load the boxes is to double-click on a data value in the Data Grid. When this is done, the address of the value is loaded in the “Addr” box and the data for that value is loaded in the “Data” box. The data can then be edited as needed.

Note it is not possible to change the data in the Data Grid directly. All data must be loaded in the “Data” box, edited there and then written back to the Data Grid or directly to the nvRAM.

The “Zero All Grid Data” will load the Data Grid with zeroes in all locations.

The “Write Byte to NVRAM” button will cause the address and data to be read from the “Addr” and “Data” boxes and write the single byte indicated by the address and data directly to nvRAM. The Data Grid is not updated with this data.

The “Read Byte from NVRAM” button will cause the address to be read from the “Addr” box and read the single byte indicated by the address directly from nvRAM. The Data Grid is not updated with this data.

The “Checksum Grid Data” button will checksum the data in the Data Grid and place this checksum in the appropriate place in the Data Grid. In order for the checksum to be computed, various PCI defined data values must already appear in the data. These items are checked before the checksum is calculated and if not present, error messages will be displayed and the checksum will not be calculated. These items are:

| Location | Data | Comments |
|---|---|---|
| | | |
| 0x00,0x0 | 0x55,0xAA | 2 byte pattern |
| 0x18,0x19 | LSB 2 bits zero | Must be DWORD boundary address |
| [0x18,0x19] | ASCII “PCIR” | At the address pointed to by the value in locations 0x18,0x19, the 4 byte pattern of “P”, “C”, “I”, and “R” must be found. |
| 0x10,0x11 | Length ID | Length indicator. The length divided by 512. |
| 0x02 | Size ID | Size indicator. The size divided by 512. |
| | | |

If all the above locations contain valid data, the checksum will be calculated and placed in the Data Grid.

The “Standard Params” button will cause another display screen to be displayed that will allow the definition of the nvRAM bytes using PCI standard and AMCC specific data. This is an alternate way to define some of the data. This screen’s functioning is described later.

The “Write All NVRAM” button will cause all the data in the Data Grid to be written to nvRAM.

The “Read All NVRAM” button will cause all the data in the nvRam to be read and displayed in the Data Grid.

**NVRAM Grid Data:**

This is used to display the data being used, whether the data was read from nvRAM, a disk file or modified manually.

**Options**:

This allows the type and size of nvRAM to be specified. The default is a Serial 24C16 type of RAM, with a capacity of 2048 bytes. This is the type normally found on the S5335DK board. Other types and sizes of nvRAM may be specified. To change the type of device, click on the arrow in the left corner of the list box to display the list (scrolling may be required to see all the types in the list) and click on the desired type from the list. The list box will be closed and the type selected will be displayed in the box.

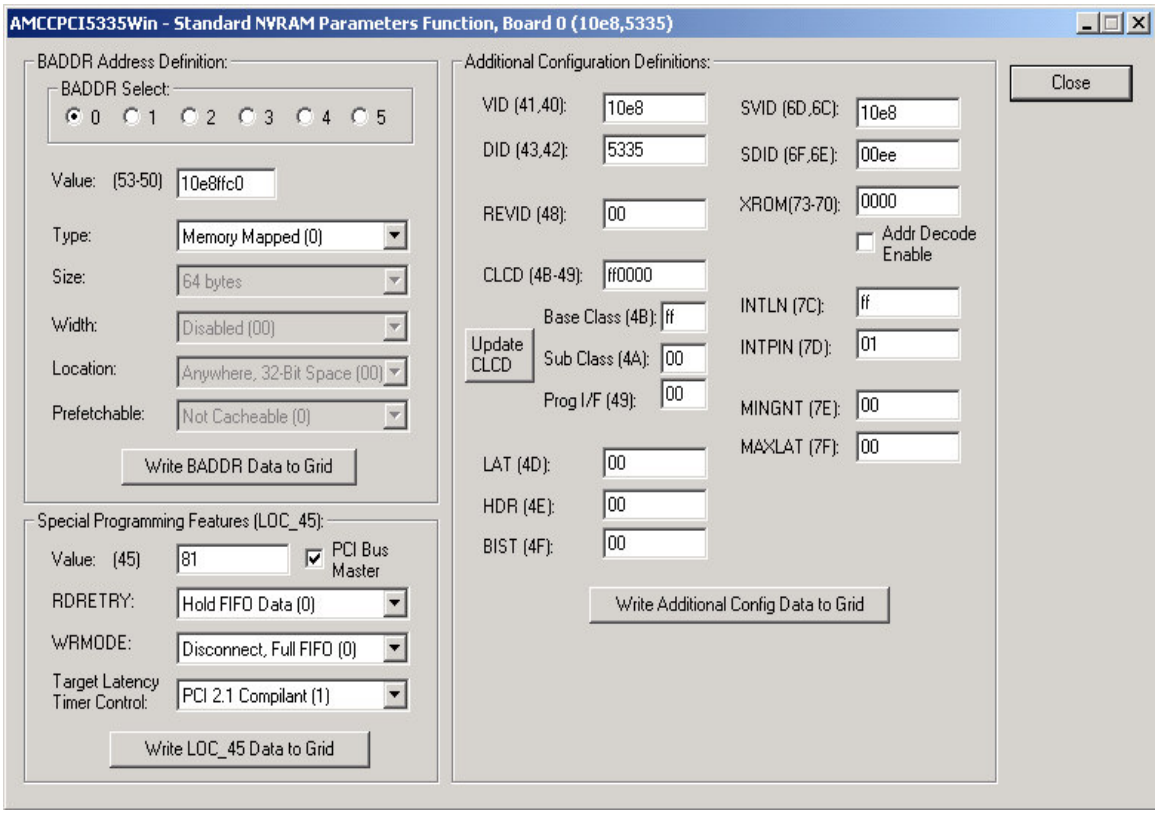
If the type of nvRAM device in use is different than the default, select the type before doing any reading or writing to a disk file, or reading or writing to nvRAM. Setting this parameter correctly ensures the proper length is used for the operations.

**Number Base:**

This allows the data in the Data Grid to be displayed in one of three number bases: Hexadecimal, Decimal or Binary (base 2). The default number base is Hex, but selecting the desired base can change it.

Note that changing the Number Base possibly also changes the number base that the Address and Data entries are made in. See that the number base for these entries is always displayed next to the boxes the data is entered in.

If the “Standard Params” button is clicked, the following display will appear:




This display allows the definition of various nvRAM data values used by either PCI or the AMCC S5335. A detailed description of each of the values displayed here can be found in the PCI Products Data Book from AMCC.

Note that the data displayed initially in this display comes directly from the Data Grid data from the NVRAM Tool display. All data changed in this display, and the corresponding “Write” button clicked, will be written to the Data Grid in the NVRAM Tool display. Data is never written from this display directly to nvRAM!

**BADDR Address Definition:**

This allows the definition of the address descriptors for BADDR0 through BADDR5 used by PCI to assign system resources I/O access to the S5335. First, select the BADDR to be defined by clicking on the correct “BADDR Select” button. This will cause the display of the current settings for this value. Then make the desired changes by selecting the entries from the various list boxes. Note that the actual data value calculated from the selections as they are made, is always displayed in the “Value” box. This box can also be edited directly.

Once the desired address descriptor defined, the “Write BADDR Data to Grid” button must be clicked to transfer the newly created data to the Data Grid. If defining more than one BADDR definition, after each definition is completed, the “Write BADDR Data to Grid” button must be clicked.

**Additional Configuration Definitions:**
This allows the definition of a number of nvRAM values used by the S5335. These values are pretty straightforward and detailed definitions of these values may be found in the PCI Products Data Book from AMCC.

Almost all of these values require editing, except for the “Addr Decode Enable” check box, which turns a single bit on (checked) or off (unchecked) in the XROM value.

The CLCD value is made up of three bytes of information. Each of these three bytes is separately editable, and to combine the separate bytes click on the “Update CLCD” button.

Once the settings are as desired, the “Write Additional Config Data to Grid” button must be clicked to transfer all of the data in this group to the Data Grid. The Data Grid is not updated as each value is changed.

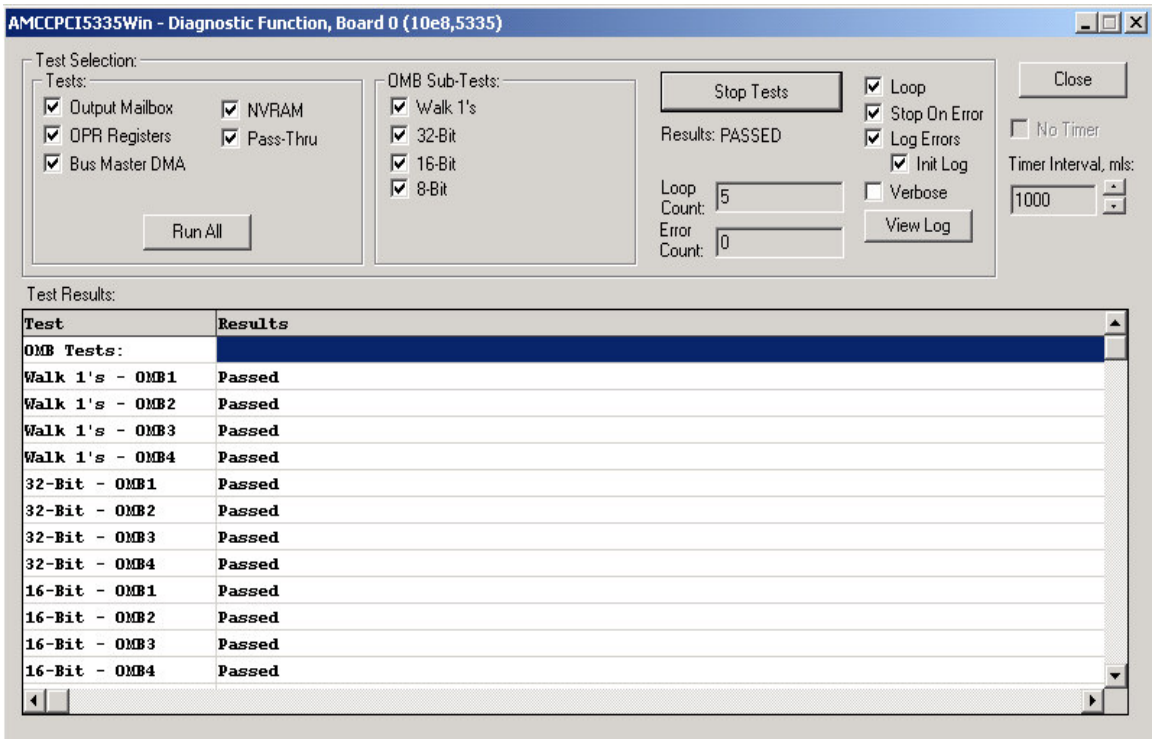
**Special Programming Features (LOC_45):**
This allows the definition of a number of nvRAM the value at location 45 that is used by the 5335 certain operational parameters. The settings for the location 45 value may be found in the PCI Products Data Book from AMCC.

Almost all of the settings for location 45 are contained in list boxes, which allow the desired setting to be selected from a list of valid settings. The “Bus Master” check box will enable (checked) or disable (unchecked) the single bus mastering bit for the device.

Once the settings are as desired, the “Write LOC_45 Data to Grid” button must be clicked to transfer all of the data in this group to the Data Grid. The Data Grid is not updated as each value is changed.

## *AMCCDIAG Diagnostics:*

This screen allows the running of diagnostics tests on the S5335DK board.




The procedure for running diagnostics is basically to select the tests, select the options for running the tests and then running the tests. The following sections describe how to do this.

**Tests/Sub-Tests:**

This allows the selection of the individual tests to run. The tests are arranged as groups of tests, with sub-tests in each group. Clicking on a group and placing a check in the check box in the “Tests” section selects that group of tests for execution. When a group is clicked, the list of tests for that group is displayed in the “Sub-Tests” section. Each of these sub-tests may be individually included (checked) or excluded (unchecked) from the group of tests. Clicking on a group in the “Tests” section and removing the check removes this entire group of tests from execution, even if the sub-test is checked.

The available tests are:

| Test Group | Sub-Tests | Description |
|---|---|---|
| | | |
| **Output Mailbox (All 4)** | | |
| | Walking 1's | Writes and reads back a one-bit pattern for each of the 32 bits of the OMB. |
| | 32-Bit | Tests writing 32 bit data patterns to the OMB. |
| | 16-Bit | Tests writing 16 bit data patterns to the OMB. Includes writing to upper and lower half of the OMB. |
| | 8-Bit | Tests writing 8 bit data patterns to the OMB. Includes writing to all 4 bytes of the OMB. |
| **OPR Registers** | | |
| | MCSR | Tests the setting of bits in the MCSR register. |
| | INTSCR | Tests the setting of all fields and bits in the INTSCR register. |
| **NVRAM** | | |
| | Loc 0 | Writes and reads data patterns to location 0 in the nvRAM. |
| | Loc 2047 | Writes and reads data patterns to location 2047 (hex 0x3ff) in the nvRAM. |
| **Pass-Thru** | | |
| | 32-Bit | Tests the writing and reading of 32-bit data patterns to the SRAM. |
| | 16-Bit | Tests the writing and reading of 16-bit data patterns to the SRAM. |
| | 8-Bit | Tests the writing and reading of 8-bit data patterns to the SRAM. |
| | All SRAM | Write and reads patterns to all 128K bytes of the SRAM. |
| **BusMaster DMA** | | |
| | PCI->AddOn | Tests the DMA operation from the PCI side to the AddOn side. |
| | AddOn->PCI | Tests the DMA operation from the AddOn side to PCI the side |

First, select all the tests that desired to be run by clicking on each group of tests and then clicking on each sub-test for that group that is to be run. If the “Run All” button is clicked, it automatically selects all Tests and Sub-Tests by putting checks in all boxes. Note that for convenience, all the tests are selected when the diagnostic screen first is displayed.

**Note: The Pass-Thru tests have a selection labeled “All Regions”. This is not a test itself, but is an indicator on how to run the Pass-Thru tests. Normally the Pass-Thru tests are run using all four regions. By un-checking the “All Regions” box, only region 1 is used for testing.**

Next, select the options for running the test. If the “Loop” box is checked, the tests selected will be run over and over again until a stop condition is met. If the “Stop On Error” box is checked, the tests will stop, even if Loop is selected, when an error is detected.

If “Log Errors” is checked, then a disk file will be written logging all errors detected. If “Init Log” is checked, and “Log Errors” is checked, then any existing log file of the same name will be erased before starting the tests for the first time. Note that “Init Log” has no effect, checked or unchecked, if “Log Errors” is not checked.

If “Verbose” is checked, then not only will error messages be displayed, but successful tests will also be displayed with information related to the passing tests.

Finally, once all the tests and run options are selected, click on “Start Tests” to begin execution of the tests. Note that if Loop has been selected, the label on the “Start Tests” button now says “Stop Tests”. Clicking on the “Stop Tests” button will stop the execution of the tests. In addition, if Loop is selected, clicking on the “Loop” check box and removing the check can stop the tests.

While the tests are running, the results of the tests are displayed in the lower grid area. This will indicate the Pass/Fail status of the tests, and if a test failed, information about the failure will be displayed. In addition, the “Loop Count” and “Error Count” will be updated indicating the number of passes of tests that have been completed and the number of error detected.

Clicking on the “View Log” will open the current log file and display it using Notepad. This can be done while the tests are running or not.

Log files are created if this option is selected. Log files will contain information about the board being tested, the time and date the tests were started and stopped and information about any errors detected. The log files are placed in the same directory as the application and they are

named as follows: “pci_5335_diag_log_#.txt”, where “#” is the board number 0, 1, etc. A separate log file may be maintained for each board running diagnostics.

**Loop Timer Ctrl:**
This allows control of the time interval between the running each pass of the selected tests. Each pass at running the selected tests is started based on a time interval, which is controlled by an internal Windows timer. The default time interval is one second, which means that the running of the next pass of tests will start one second after the previous pass started. This time interval can be adjusted up or down at 50 millisecond intervals by using the arrow keys to the right of the Timer Interval display box. The interval can be set anywhere from 50 milliseconds to 2 seconds (2000 milliseconds) at 50 millisecond increments.

The use of timers has the advantage of allowing multiple boards to be tested automatically by allowing CPU control to be shared between the diagnostic windows for each board.

There is one additional check box control that stops the use of the timers and times the amount of time between each pass of tests internally to the program. This was used at one time for debugging, but is of little use now.

# Chapter 5
# AMCCPCI5335AddOnWin Software Application

The AMCCPCI5335AddOnWin application is a Windows 98/2000 application that can do the following:

- Display Configuration and Operational registers
- Allow the Operational Registers to be modified
- Modify and Program the nvRAM
- Run Diagnostics

Besides the Diagnostics used for System Checkout discussed previously, there are a number of other screens and features contained in the AMCCPCI5335AddOnWin application. These are explained in the following section.

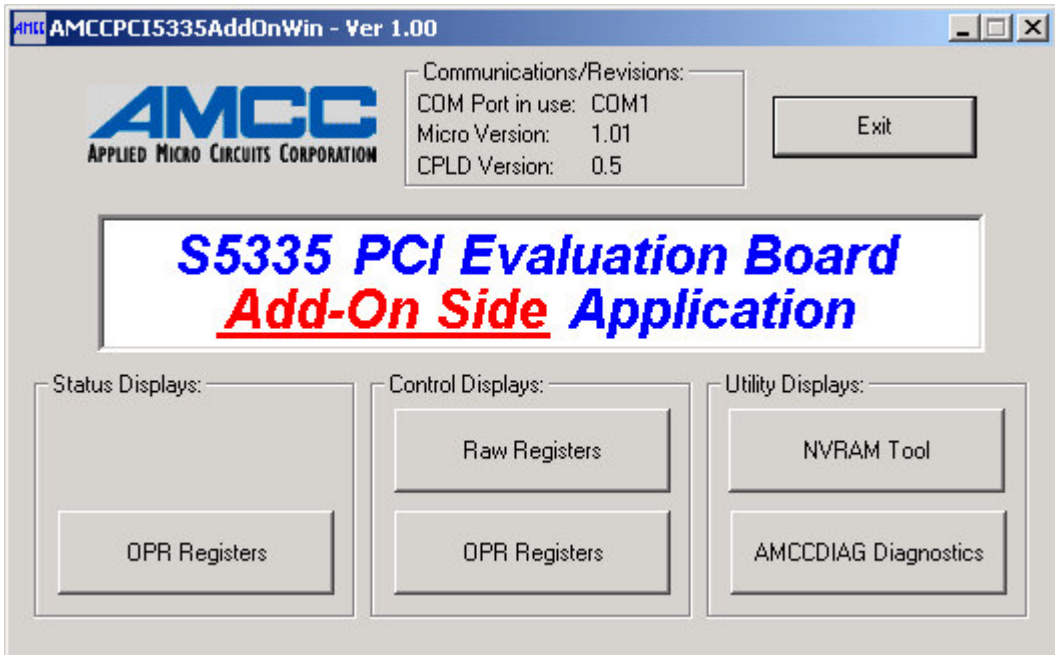
When the AMCCPCI5335AddOnWin is loaded, it will search for the S5335DK board on either COM1 or COM2 and display the COM port in use on the main screen. In addition, the versions of the ATMEL micro code and CPLD code are displayed on the main screen for informational purposes.

There is only one S5335DK board that can be connected to the application.

Note that source code for this application is NOT supplied on CD. Source code is supplied for a smaller version of this application that contains only the NVRAM Utility portion of this application. This smaller application is AMCCNVRAMWIN. The source code contains complete code for the application, library and driver to support the nvRAM access only.

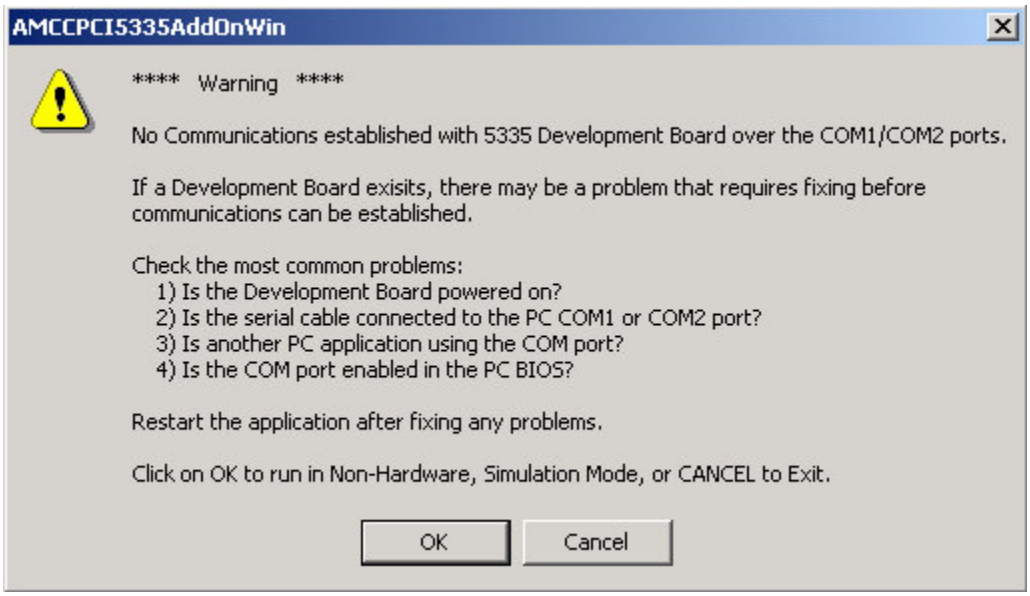
## Main Screen

This screen is the first screen displayed after the AMCCPCI5335AddOnWin application is started.




This screen is divided into 3 areas. The “Status Displays” will display static information from the registers. The “Control Displays” will allow the settings in the registers to be changed. Finally, the “Utility Displays” will allow other settings and values to be changed on the board.

Note that if no PCI card is detected, the following error message may be displayed and the AMCCPCI5335AddOnWin application will not run properly:




This indicates that there is a problem in communicating from the PC to the S5335DK board over the RS-232 COM port. Check the list in the message for hints to correcting the problem.

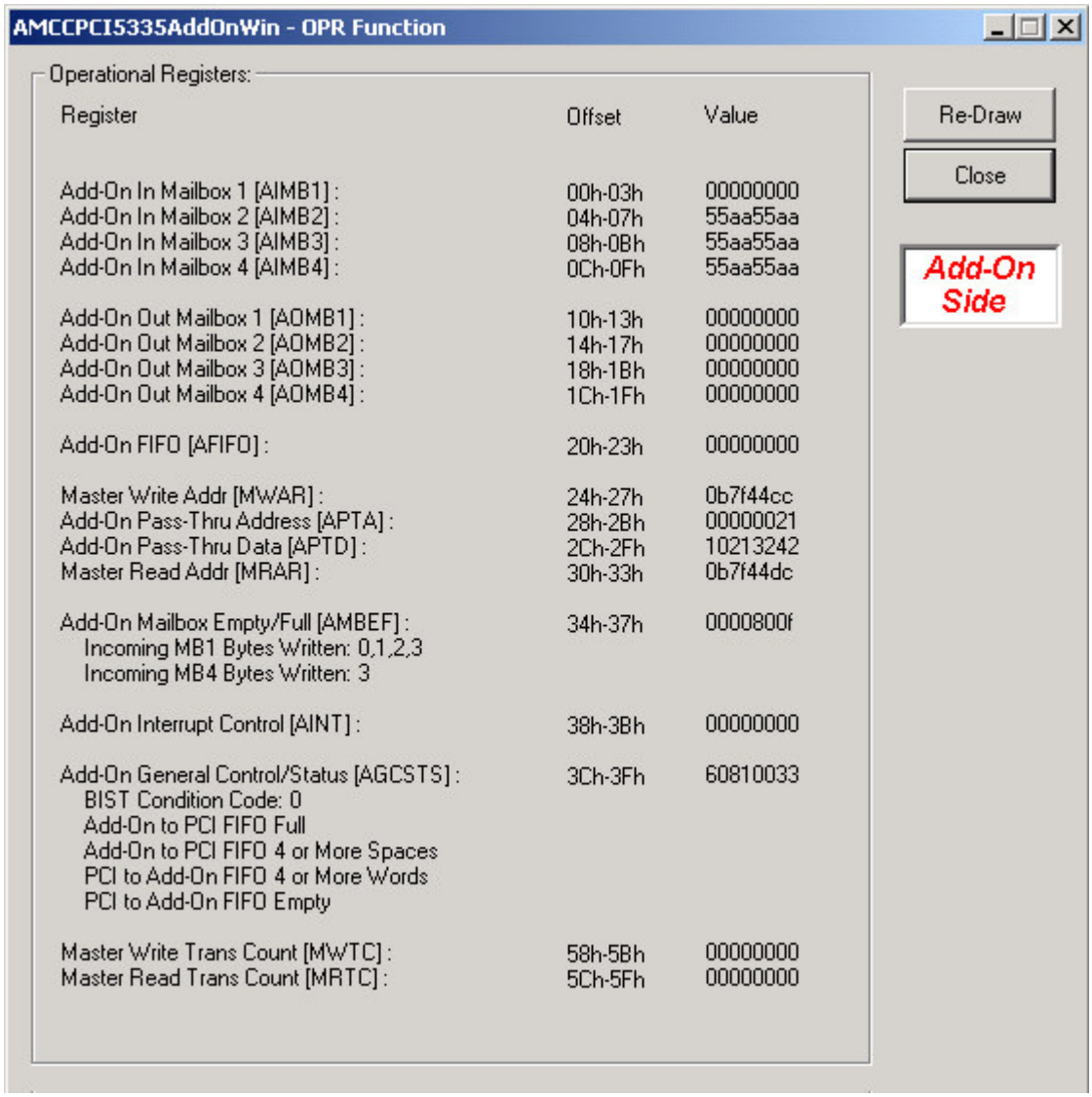
Clicking on “OK” here will run the AMCCPCI5335AddOnWin application in simulation mode, where it does NOT communicate with the S5335DK board. It simulates the S5335 operational registers using internal arrays. Clicking on “Cancel” will terminate the program.

## *Status Displays*

### OPR Registers:

This static display shows the Operational Register values for the board selected. The board is selected from the front screen before this screen is displayed. Note that the screen’s menu bar displays the board number. Clicking on the “Re-Draw” button will re-read the Operational Registers for the board and display the results again.

The image below shows a typical Operational Register display. It shows the Register name, the Offset address and the current Value for the registers. In addition, it shows the state of some of the bits within the registers.

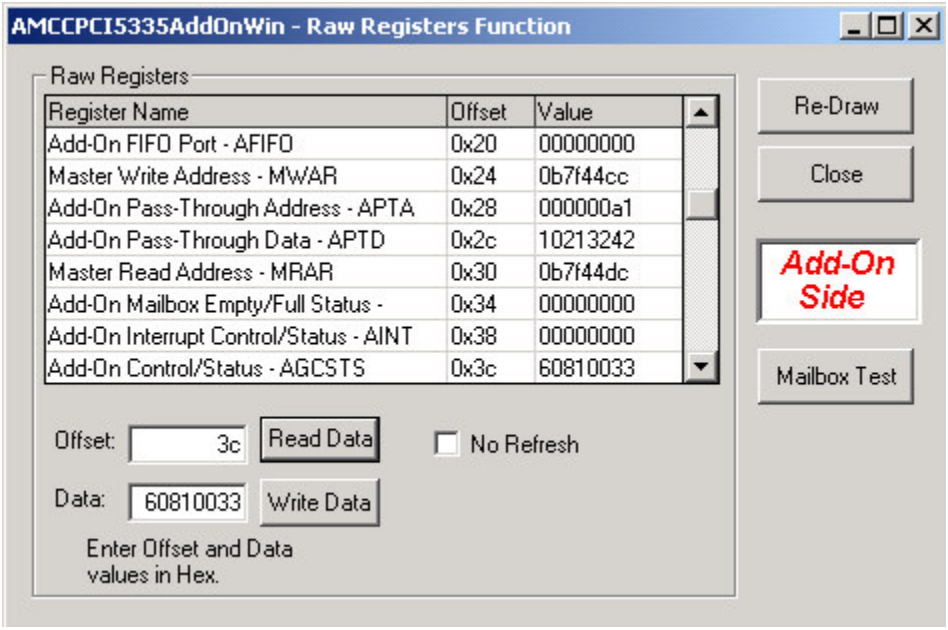



Note that this is a static display and no Operational Register values can be changed from this screen. See the OPR Control screen description for changing Operational Register values.

## Control Displays

### Raw Registers:

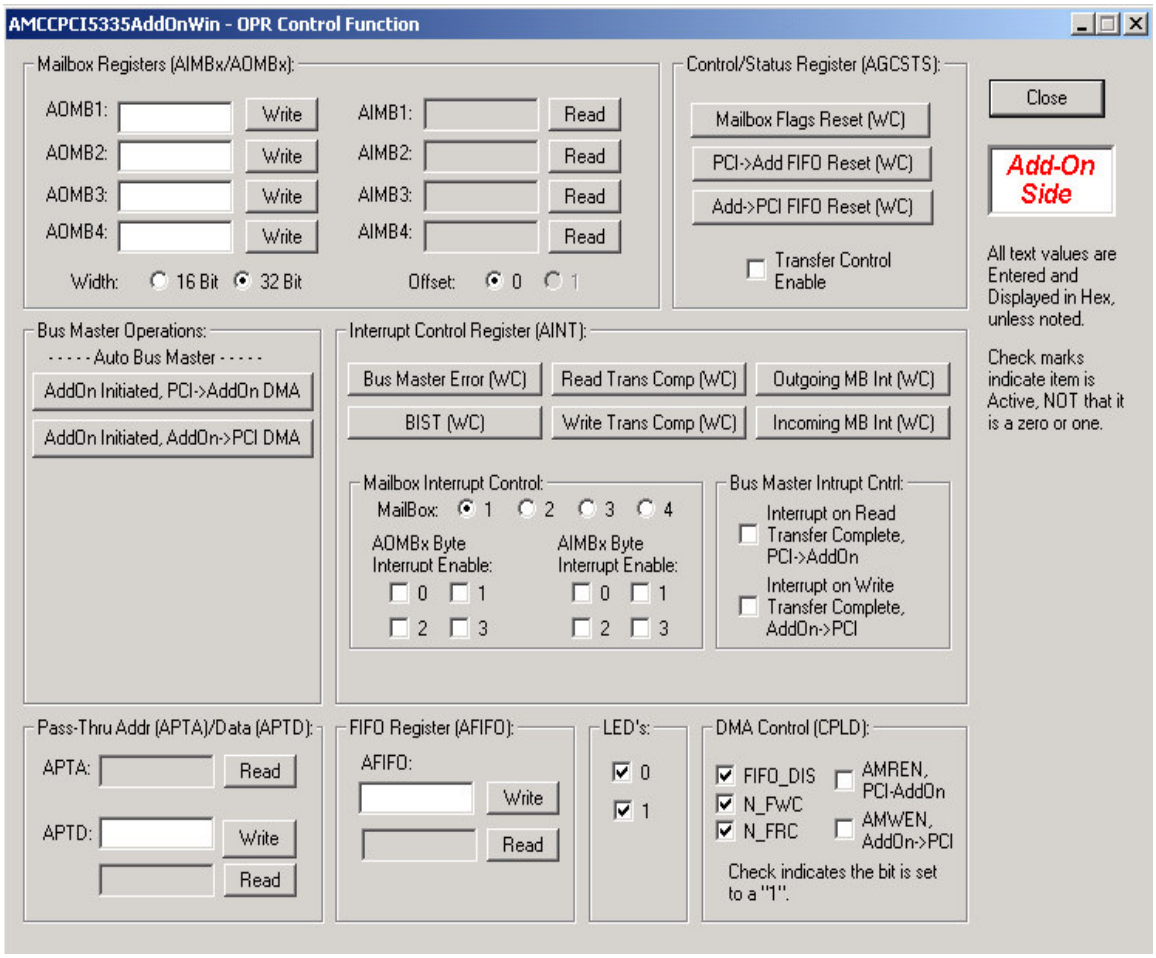
This screen allows the programming of the Operational Registers by reading and writing full 32-bit registers.




This screen consists of a list of the registers showing the values for the registers in the grid area. By entering the “Offset” and clicking the “Read Data: button, the register at the offset will be read and displayed in the “Data” box, as well as in the grid. In addition, by entering the “Offset” and “Data” values in the respective boxes, and clicking the “Write Data” button, the data entered will be written to the register.

**OPR Registers:**

This screen allows the programming of the Operational Registers to the settings required for a particular test or usage of the board by programming the individual functional fields within the registers.




This screen consists of many buttons and controls that easily allow the setting of the Operational Register controls. There are basically four types of controls on this screen: buttons, check boxes, data entry boxes and list boxes containing a list of valid selections.

When clicked, the buttons cause an action to take place. This action may be the writing of a single bit (“FIFO Reset (WO)”, for example), the reading of a text entry box and writing this value to a register (“Write” button in the SRAM controls), or a sequence of steps to carry out an action (“Cycle Add-On Reset”).

The check boxes allow the setting or clearing of a single bit control. When clicked and the check box has a check mark, the indicated control is on, or active. When clicked and the check box does not have a check mark, the control is off, or inactive.

The data entry boxes allow a value to be typed and when an associated “Write” or “Read” button is clicked, the information contained in this box is used for the operation. No action is taken to use any of the data entered until the associated button is clicked to start the action. Note that there are some boxes that white and others that are gray, or darkened. Only in the white boxes can data be entered. The gray boxes are read only and are used to display values only.

The list boxes operate very much like the data entry boxes and their associated button. The difference is that for the controls that use list boxes, there are only a small number of correct entries and we list them all in the list box. When the arrow in the box is clicked, all the items in the list are displayed (scrolling may be required to see them all), and the desired entry should be clicked. The list box will then be closed with the item selected displayed, and the value that corresponds to the item selected will be written to the register.

Note that anytime a change is made using this control screen, an update command is sent to the OPR Registers status display, if it is displayed. Any changes, therefore, to the OPR registers should be automatically displayed in the status window.

**Mailbox Bytes (AOMBx/AIMBx):**

These controls allow the writing of the four AOMB registers and the reading of the four AIMB registers. First select the bit width of the operation, 16-bit or 32-bit, and select the offset associated with the bit width of the operation. Note that the available offsets will change as the bit width is changed. For a write to the AOMB register, enter the data to be written in the box for the register. Then click the corresponding “Write” button to write to the AOMB register. To read, click the “Read” button for the register to read from the AIMB register. And display the data read in the corresponding box.

**AddOn Interrupt Control Register (AINT):**

These controls allow the writing of the individual fields of the Interrupt Control Register, or INTSCR. The two buttons, “Clear Incoming MB Int” and “Clear Outgoing MB Int” will write a one to the Incoming Mailbox Interrupt bit, or the Outgoing Mailbox Interrupt bit, respectively. The “Add-On Interrupt” check box enables (checked) or disables (not checked) the Add-On Interrupt Pin (ADDINT#) Enable signal.

The two groups of four check boxes for OMB Byte Interrupt Enable and IMB Byte Interrupt Enable, will enable (checked) or disable (not checked) the interrupt enable by building the command and issuing it to the Interrupt Control Register.

**AddOn General Control/Status Register (AGCSTS):**

These controls allow the writing of the individual fields of the Reset Control Register, or RCR. The two buttons, “FIFO Reset (WO)” and “Mailbox Flags

Reset (WO)" will write a one to the Read FIFO Reset bit, or the Mailbox Flags Reset bit, respectively.

The "Add-On Reset" check box enables (checked) or disables (not checked) the Add-On Reset bit. The "Cycle Add-On Rest" button will automatically cycle the Add-On Reset bit by enabling it, then disabling it.

**Interrupt Control/Status Register (AINT):**
These controls allow the writing of the individual fields of the Interrupt Control Register, or INTSCR. The six buttons, such as "Bus Master Error (WC)" and "Read Trans Comp (WC)" will write a one to the corresponding interrupt bit.

The two groups of four check boxes for AOMB Byte Interrupt Enable and AIMB Byte Interrupt Enable, will enable (checked) or disable (not checked) the interrupt enable by building the command and issuing it to the Interrupt Control Register.

Finally, the two check boxes to set or clear the read and write transfer complete bits.

**Bus Master Operations:**
These controls allow the user to initiate Bus Master DMA operations from the AddOn side. DMA operations can move data to or from the PCI side and AddOn side.

The AddOn side Bus Mastering is a somewhat manual operation that requires significant help from the PCI side operation for each DMA operation done. Both applications must be running to successfully complete an AddOn side initiated DMA operation.

The physical address of the memory buffer, the byte count, the generation of data patterns and the checking of data after the DMA is complete are done by the PCI side application. The physical address of the memory buffer and the byte count are passed to the AddOn side application to be loaded in registers for the DMA operation.

The steps in completing an AddOn side Bus Mastering DMA operation are as follows:

1) On the AddOn side application, click on either the

   "AddOn Initiated, PCI->AddOn DMA" or
   "AddOn Initiated, AddOn->PCI DMA"

   button to begin the desired DMA operation.

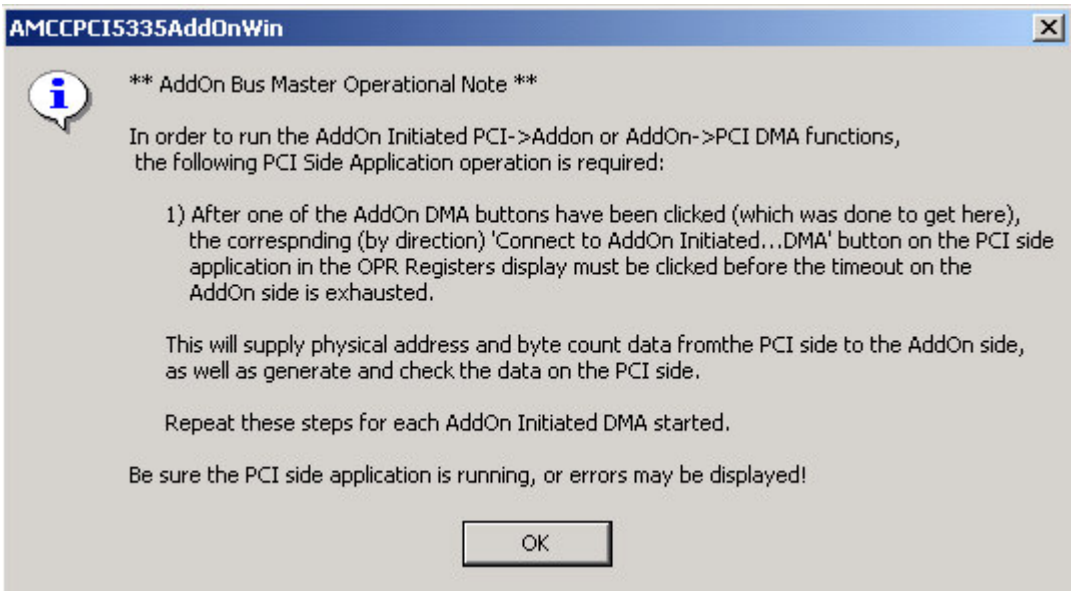
2) On the PCI side application, click on the corresponding

"Connect to AddOn Initiated , PCI->AddOn DMA" or
"Connect to AddOn Initiated , AddOn->PCI DMA"

button. This will cause the initial handshaking to take place.

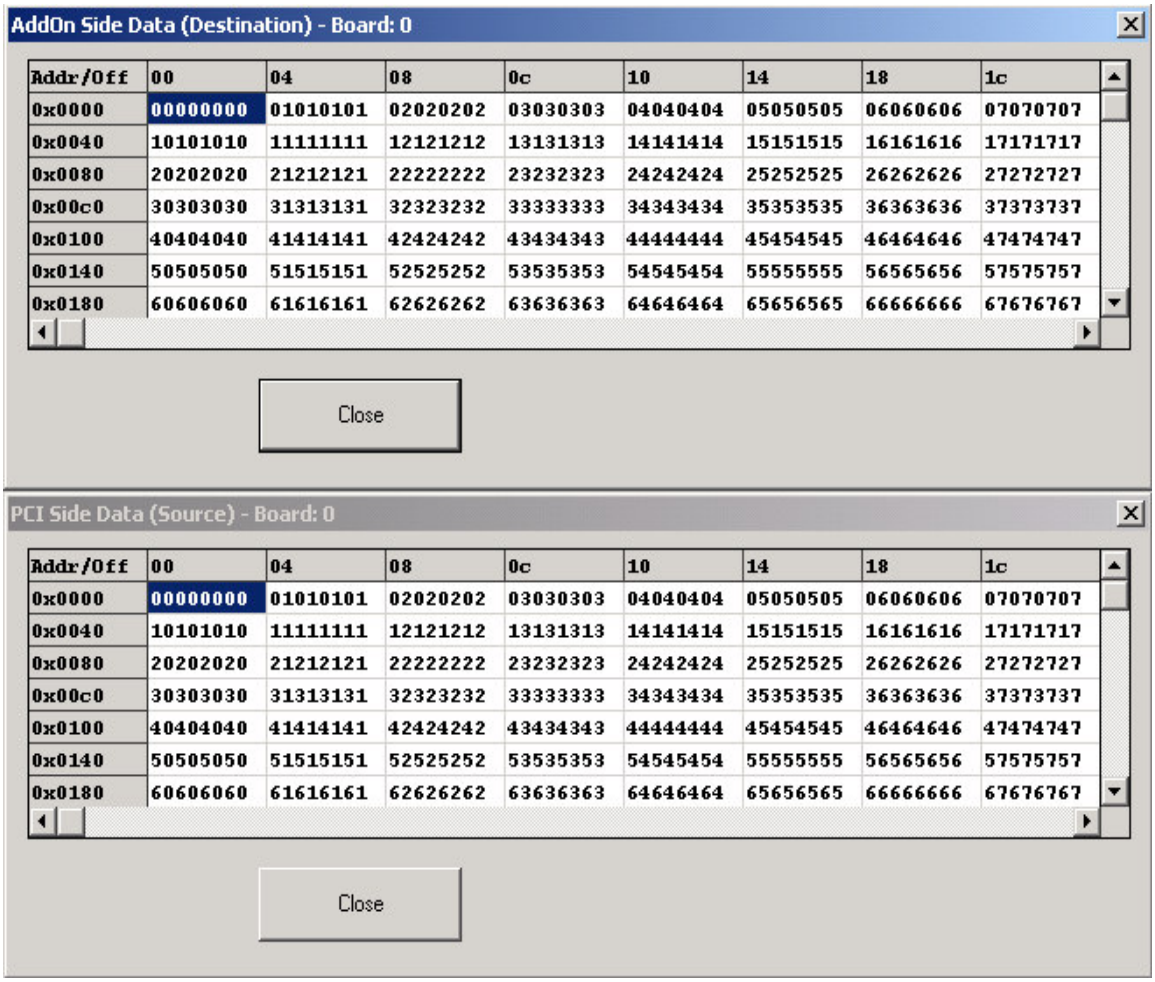
At this time the DMA transfer will complete and the data will be checked by the PCI side application. Any data errors detected, as well as the data buffers, will be displayed by the PCI side application.

On the AddOn side application, the user will be reminded of this handshaking each time the first DMA operation is attempted with the following message:




Click on the OK button on this message before going to the PCI side application to click the button there.

The data displayed by the PCI side application will be in a format similar to the following, showing the Source and Destination data:




**FIFO Register (AFIFO):**
These controls allow the reading and writing of the FIFO register.

To write to the FIFO, enter the 32-bit data value in the box, in hex, and click on the “Write” button.

To read from the FIFO, click on the “Read” button. The data read will be displayed in the gray box.

**LED’s:**
These controls allow the setting of two of the LED’s located on the S5335DK board. Place a check in the check box to turn the LED on, or uncheck the box to turn it off.

**DMA Control (CPLD):**
These controls allow the setting of the FIFO_DIS control to the CPLD, which is used for PCI side initiated Bus Master operations, or four FIFO control signals: N_FWC (inverted FWC#), N_FRC (inverted FRC#), AMREN and AMWEN.

Place a check on these boxes to set the bit to a one, or uncheck the box to set the bit to a zero.

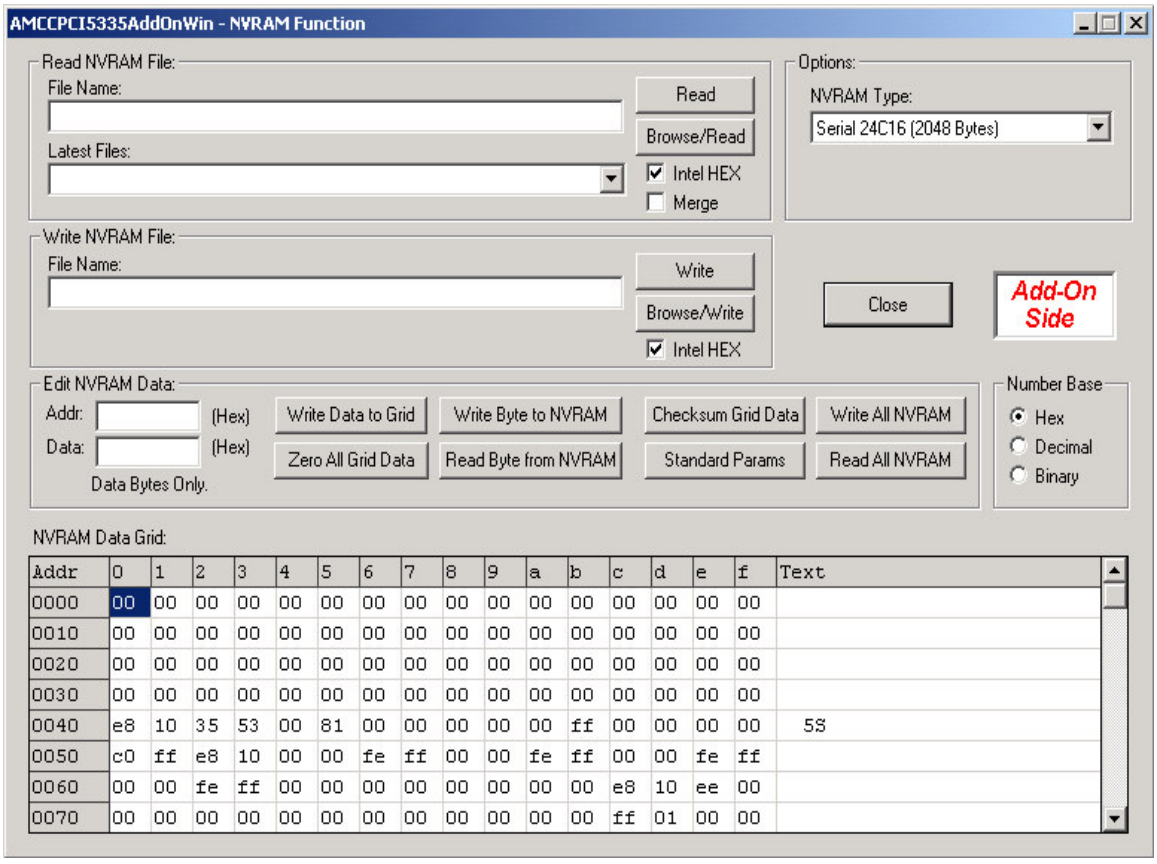
**Note: The two "AddOn Iniated->…" buttons as well as the N_FWC, N_FRC, AMREN and AMWEN controls are NOT available and will be grayed out if AddOn Bus Mastering is NOT enabled from the nvRAM.**

## Utility Displays

### NVRAM Tool:

This screen allows the building of an nvRAM image and/or the reading and writing of an image from/to the nvRAM.

**Note: It may take a minute to read the contents of nvRAM from the AddOn side and display this screen.**



When the screen is first brought up, the existing nvRAM data from the board is read and displayed in the Data Grid area. So this will first show the existing nvRAM data. After this, a disk file may be read with new data, the data may be modified, written back to the nvRAM or a disk file written with the data.

**Read NVRAM File:**

This section allows the reading of a disk file into the Data Grid. First, set the parameters for the file by indicating if the file is an Intel Hex format file (check the “Intel HEX” box) or is a binary file (uncheck the “Intel HEX” box). An Intel Hex file has a specific ASCII format to it, while a binary file is treated as bytes of data, one byte in the file is one byte of nvRAM data.

Next, indicate whether the data in the file is to be overlaid on the existing data in the Data Grid (check the “Merge” box), or should the data in the Data Grid be cleared (uncheck the “Merge” box) before the data is loaded.

Next, the file needs to be selected. The name of a file to read can be typed into the “File Name” box and then read by clicking on the “Read” button, but most often the “Browse/Read” button is going to be clicked which will bring up a file browser window. This browser window can then be used to navigate to the desired file. From the browser box, the file can be doubled clicked to load it into the Data Grid, or can be highlighted (single clicked) and the “Open” button clicked to load the file.

At this point, the file is opened, the data is read and the data is displayed in the Data Grid area.

Note that the file name read is now displayed in the “File Name” box. This box always displays the last file read. If this file is to be read again, just click on the “Read” button and the file will automatically be read without having to browse again.

Additionally, as each file is read, the name is placed in the “Latest Files” box. This keeps a list of all files read. If a file needs to be re-read, and it is not the last file read, pick the file from this list and it will be loaded automatically without browsing. Just make sure the “Intel HEX” and “Merge” boxes are set correctly.

**Write NVRAM File:**

Writing a file to disk is very similar to reading a file, but in this case the data from the Data Grid is written to a disk file. . First, set the parameter for the file by indicating if the file is to be an Intel Hex format file (check the “Intel HEX” box) or is to be a binary file (uncheck the “Intel HEX” box).

Next, the file needs to be selected. Again, the name of a file to write can be typed into the “File Name” box and then written by clicking on the “Write”

button, but most often the “Browse/Write” button is going to be clicked which will bring up a file browser window. This browser window can then be used to navigate to the desired file or directory. From the browser box, the file can be doubled clicked to write the data from the Data Grid, or can be highlighted (single clicked) and the “Open” button clicked to write the file. In either of these cases, the existing file will be over written and the original contents will be lost.

If the file is a new file, then navigate to the directory for the file, type the name of the file in the box provided and then click on “Open”. The new file will be opened and written with the data.

**Edit NVRAM Data:**

This is used to modify the data in the Data Grid or read and write data to the nvRAM. The data bytes in the Data Grid can only be edited using the “Addr” and “Data” boxes. Type the address of the data in the “Addr” box and place the new data value in the “Data” box. Note that the number base required for each entry is displayed just to the right of each box in parentheses. Once the address and data have been entered, the other buttons can be used to do something with them.

Rather than typing the address and data information into the boxes, an alternate way to load the boxes is to double-click on a data value in the Data Grid. When this is done, the address of the value is loaded in the “Addr” box and the data for that value is loaded in the “Data” box. The data can then be edited as needed.

Note it is not possible to change the data in the Data Grid directly. All data must be loaded in the “Data” box, edited there and then written back to the Data Grid or directly to the nvRAM.

The “Zero All Grid Data” will load the Data Grid with zeroes in all locations.

The “Write Byte to NVRAM” button will cause the address and data to be read from the “Addr” and “Data” boxes and write the single byte indicated by the address and data directly to nvRAM. The Data Grid is not updated with this data.

The “Read Byte from NVRAM” button will cause the address to be read from the “Addr” box and read the single byte indicated by the address directly from nvRAM. The Data Grid is not updated with this data.

The “Checksum Grid Data” button will checksum the data in the Data Grid and place this checksum in the appropriate place in the Data Grid. In order for the checksum to be computed, various PCI defined data values must already appear in the data. These items are checked before the checksum is

calculated and if not present, error messages will be displayed and the checksum will not be calculated. These items are:

| Location | Data | Comments |
|---|---|---|
| | | |
| 0x00,0x0 | 0x55,0xAA | 2 byte pattern |
| 0x18,0x19 | LSB 2 bits zero | Must be DWORD boundary address |
| [0x18,0x19] | ASCII “PCIR” | At the address pointed to by the value in locations 0x18,0x19, the 4 byte pattern of “P”, “C”, “I”, and “R” must be found. |
| 0x10,0x11 | Length ID | Length indicator. The length divided by 512. |
| 0x02 | Size ID | Size indicator. The size divided by 512. |
| | | |

If all the above locations contain valid data, the checksum will be calculated and placed in the Data Grid.

The “Standard Params” button will cause another display screen to be displayed that will allow the definition of the nvRAM bytes using PCI standard and AMCC specific data. This is an alternate way to define some of the data. This screen’s functioning is described later.

The “Write All NVRAM” button will cause all the data in the Data Grid to be written to nvRAM.

The “Read All NVRAM” button will cause all the data in the nvRam to be read and displayed in the Data Grid.

**NVRAM Grid Data:**

This is used to display the data being used, whether the data was read from nvRAM, a disk file or modified manually.

**Options**:

This allows the type and size of nvRAM to be specified. The default is a Serial 24C16 type of RAM, with a capacity of 2048 bytes. This is the type normally found on the S5335DK board. Other types and sizes of nvRAM may be specified. To change the type of device, click on the arrow in the left corner of the list box to display the list (scrolling may be required to see all the types in the list) and click on the desired type from the list. The list box will be closed and the type selected will be displayed in the box.

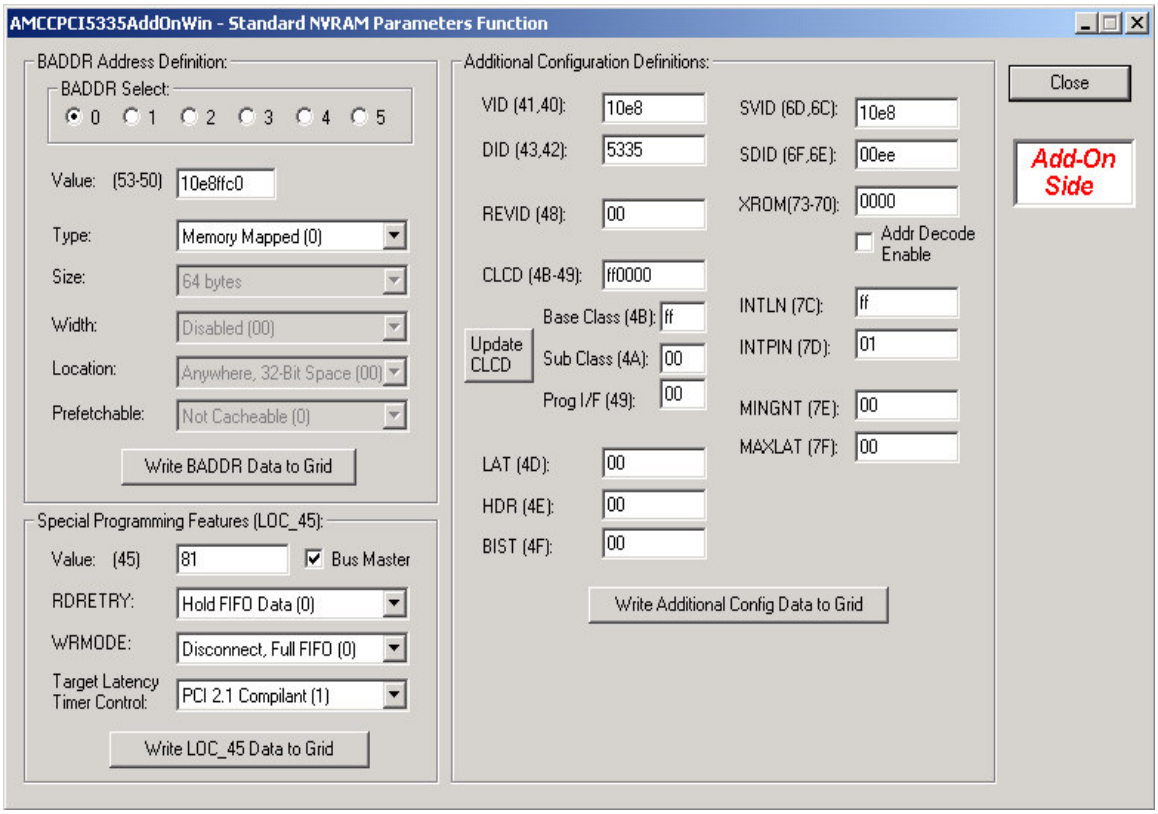
If the type of nvRAM device in use is different than the default, select the type before doing any reading or writing to a disk file, or reading or writing to nvRAM. Setting this parameter correctly ensures the proper length is used for the operations.

**Number Base:**

This allows the data in the Data Grid to be displayed in one of three number bases: Hexadecimal, Decimal or Binary (base 2). The default number base is Hex, but selecting the desired base can change it.

Note that changing the Number Base possibly also changes the number base that the Address and Data entries are made in. See that the number base for these entries is always displayed next to the boxes the data is entered in.

If the “Standard Params” button is clicked, the following display will appear:




This display allows the definition of various nvRAM data values used by either PCI or the AMCC S5335. A detailed description of each of the values displayed here can be found in the PCI Products Data Book from AMCC.

Note that the data displayed initially in this display comes directly from the Data Grid data from the NVRAM Tool display. All data changed in this display, and the corresponding “Write” button clicked, will be written to the Data Grid in the NVRAM Tool display. Data is never written from this display directly to nvRAM!

**BADDR Address Definition:**
This allows the definition of the address descriptors for BADDR0 through BADDR5 used by PCI to assign system resources I/O access to the

S5335. First, select the BADDR to be defined by clicking on the correct "BADDR Select" button. This will cause the display of the current settings for this value. Then make the desired changes by selecting the entries from the various list boxes. Note that the actual data value calculated from the selections as they are made, is always displayed in the "Value" box. This box can also be edited directly.

Once the desired address descriptor defined, the "Write BADDR Data to Grid" button must be clicked to transfer the newly created data to the Data Grid. If defining more than one BADDR definition, after each definition is completed, the "Write BADDR Data to Grid" button must be clicked.

**Additional Configuration Definitions:**
This allows the definition of a number of nvRAM values used by the S5335. These values are pretty straightforward and detailed definitions of these values may be found in the PCI Products Data Book from AMCC.

Almost all of these values require editing, except for the "Addr Decode Enable" check box, which turns a single bit on (checked) or off (unchecked) in the XROM value.

The CLCD value is made up of three bytes of information. Each of these three bytes is separately editable, and to combine the separate bytes click on the "Update CLCD" button.

Once the settings are as desired, the "Write Additional Config Data to Grid" button must be clicked to transfer all of the data in this group to the Data Grid. The Data Grid is not updated as each value is changed.

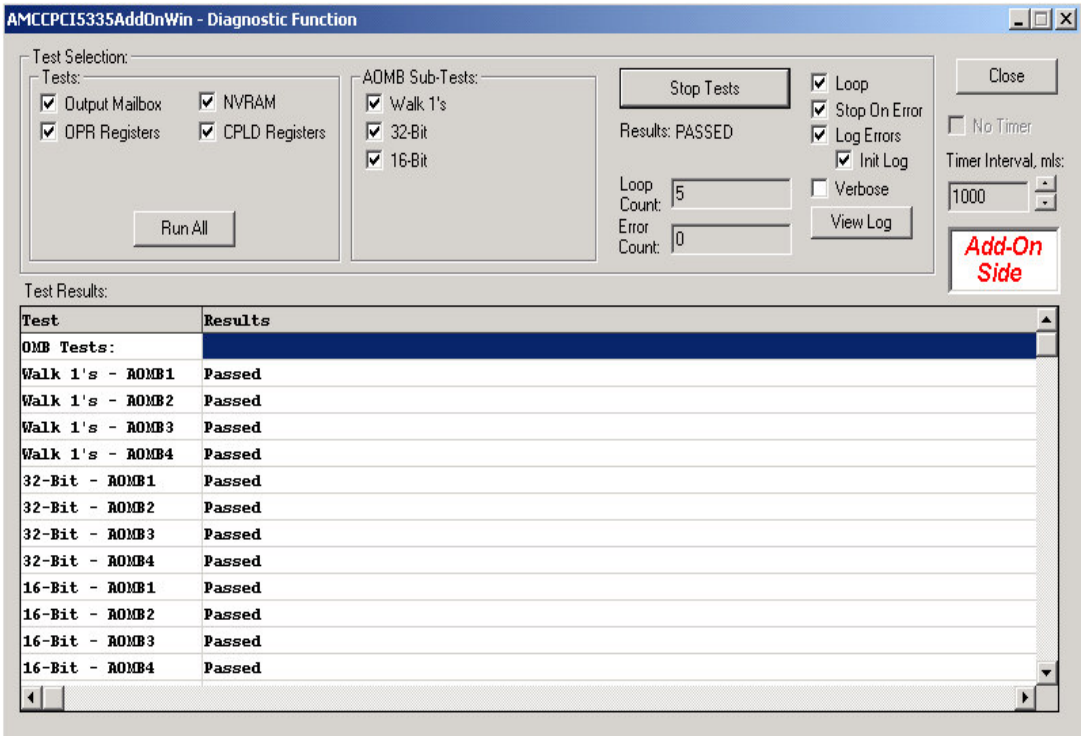
**Special Programming Features (LOC_45):**
This allows the definition of a number of nvRAM the value at location 45 that is used by the 5335 certain operational parameters. The settings for the location 45 value may be found in the PCI Products Data Book from AMCC.

Almost all of the settings for location 45 are contained in list boxes, which allow the desired setting to be selected from a list of valid settings. The "Bus Master" check box will enable (checked) or disable (unchecked) the single bus mastering bit for the device.

Once the settings are as desired, the "Write LOC_45 Data to Grid" button must be clicked to transfer all of the data in this group to the Data Grid. The Data Grid is not updated as each value is changed.

## *AMCCDIAG Diagnostics:*

This screen allows the running of diagnostics tests on the S5335DK board.




The procedure for running diagnostics is basically to select the tests, select the options for running the tests and then running the tests. The following sections describe how to do this.

**Tests/Sub-Tests:**
This allows the selection of the individual tests to run. The tests are arranged as groups of tests, with sub-tests in each group. Clicking on a group and placing a check in the check box in the “Tests” section selects that group of tests for execution. When a group is clicked, the list of tests for that group is displayed in the “Sub-Tests” section. Each of these sub-tests may be individually included (checked) or excluded (unchecked) from the group of tests. Clicking on a group in the “Tests” section and removing the check removes this entire group of tests from execution, even if the sub-test is checked.

The available tests are:

| Test Group | Sub-Tests | Description |
|---|---|---|
| | | |
| **Output Mailbox** | | |
| | Walking 1's | Writes and reads back a one-bit pattern for each of the 32 bits of the OMB. |
| | 32-Bit | Tests writing 32 bit data patterns to the OMB. |
| | 16-Bit | Tests writing 16 bit data patterns to the OMB. Includes writing to upper and lower half of the OMB. |
| **OPR Registers** | | |
| | AGCSTS | Tests the setting of bits in the AGCSTS register. |
| | AINT | Tests the setting of all fields and bits in the AINT register. |
| **NVRAM** | | |
| | Loc 0 | Writes and reads data patterns to location 0 in the nvRAM. |
| | Loc 2047 | Writes and reads data patterns to location 2047 (hex 0x3ff) in the nvRAM. |
| **CPLD Registers** | | |
| | Version | Reads the version register. |
| | Control | Tests the writing and reading of the bits in the Control register. |
| | Status | Tests the reading of the Status register. |
| | ATMEL LED | Writes and reads the LED register. |

First, select all the tests that desired to be run by clicking on each group of tests and then clicking on each sub-test for that group that is to be run. If the “Run All” button is clicked, it automatically selects all Tests and Sub-Tests by putting checks in all boxes. Note that for convenience, all the tests are selected when the diagnostic screen first is displayed.

Next, select the options for running the test. If the “Loop” box is checked, the tests selected will be run over and over again until a stop condition is met. If the “Stop On Error” box is checked, the tests will stop, even if Loop is selected, when an error is detected.

If “Log Errors” is checked, then a disk file will be written logging all errors detected. If “Init Log” is checked, and “Log Errors” is checked, then any existing log file of the same name will be erased before starting the tests for the first time. Note that “Init Log” has no effect, checked or unchecked, if “Log Errors” is not checked.

If “Verbose” is checked, then not only will error messages be displayed, but successful tests will also be displayed with information related to the passing tests.

Finally, once all the tests and run options are selected, click on “Start Tests” to begin execution of the tests. Note that if Loop has been selected, the label on the “Start Tests” button now says “Stop Tests”. Clicking on the “Stop Tests” button will stop the execution of the tests. In addition, if Loop is selected, clicking on the “Loop” check box and removing the check can stop the tests.

While the tests are running, the results of the tests are displayed in the lower grid area. This will indicate the Pass/Fail status of the tests, and if a test failed, information about the failure will be displayed. In addition, the “Loop Count” and “Error Count” will be updated indicating the number of passes of tests that have been completed and the number of error detected.

Clicking on the “View Log” will open the current log file and display it using Notepad. This can be done while the tests are running or not.

Log files are created if this option is selected. Log files will contain information about the board being tested, the time and date the tests were started and stopped and information about any errors detected. The log files are placed in the same directory as the application and they are named as follows: “pci_5335_diag_log_#.txt”, where “#” is the board number 0, 1, etc. A separate log file may be maintained for each board running diagnostics.

**Loop Timer Ctrl:**
This allows control of the time interval between the running each pass of the selected tests. Each pass at running the selected tests is started based on a time interval, which is controlled by an internal Windows timer. The default time interval is one second, which means that the running of the next pass of tests will start one second after the previous pass started. This time interval can be adjusted up or down at 50 millisecond intervals by using the arrow keys to the right of the Timer Interval display box. The interval can be set anywhere from 50 milliseconds to 2 seconds (2000 milliseconds) at 50 millisecond increments.

The use of timers has the advantage of allowing multiple boards to be tested automatically by allowing CPU control to be shared between the diagnostic windows for each board.

There is one additional check box control that stops the use of the timers and times the amount of time between each pass of tests internally to the program. This was used at one time for debugging, but is of little use now.

# Chapter 6
# AMCCNVRAMWin Software Application (Source Code Included)

Included on the CD is the source code for an application that contains only the nvRAM portion of the AMM5335PCIWin. It is called AMCCNVRAMWin. This application includes source code for the following:

- An executable containing the screens required for nvRAM reading, writing and modification
- A Windows Dynamic Linked Library (DLL) for interfacing between the executable and the driver
- A Windows PCI driver for the S5335

Additional required tools for compiling the source code:

- Microsoft Visual C/C++, Version 6.0
- Windows 98/2000/Server 2003 Device Driver Kit (DDK)

****** Caution ******

Note that the source code contained here does not fully support all capabilities and features of the S5335. In fact, they are restricted function versions of software that support only those functions required to identify the boards and write and read the nvRAM. Therefore, the driver and DLL created for this application are not usable by the AMM5335PCIWin application.

For a complete set of source code, contact AMCC and AMCC partners.

# Chapter 7
# Hardware

## S5335 PCI Card

The following section describes various aspects of the hardware design for the S5335 PCI card.

### Jumper Descriptions

***JP1 (MODE).*** MODE defines the Add-On bus DQ width. JP1 shorted configures the DQ bus for 32 bits and open configures a 16 bit DQ bus. The default setting is open for developer kit SRAM operation so that the GUI/µC can change this as needed.

***JP2 (SNV).*** SNV defines serial nvRAM or Parallel nvRAM operation. JP2 shorted configures parallel nvRAM and open configures serial nvRAM. The developer board was designed for serial nvRAM operation so JP2 should be open for normal operation.

***JP3 (ICC).*** JP3 connects the 3.3V supply to the S5335 device. It allows the designer to measure the device current. JP3 must be installed for normal operation. Refer to the S5335 PCI card schematic before altering the default configuration.

***JP4, JP5, JP6 (GND).*** These are ground jumpers to provide ground reference for scope probes and logic analyzer or other test equipment.

### Test Point Descriptions

***TP1 - PCI CLK.*** General purpose test point of the PCI clock. To be temporarily used with a low impedance oscilloscope probe for examine PCI clock signal integrity.

***TP2 - IRQ#.*** General purpose test point of the Add-On bus interrupt output from the PCI bus. To be used to examine by an oscilloscope or logic analyzer to see the presence of an interrupt from the PCI bus to the Add-On bus.

***TP3 - BPCLK.*** General purpose test point of the S5335's Add-On bus BPCLK's synchronous clock output. To be temporarily used with a low impedance oscilloscope probe for examine BPCLK's output signal integrity.

## PCB Connector Descriptions

***J1 thru J6*** - The Add-On bus from the S5335 is wired to an on board CPLD and SRAM application design. The Add-On bus is also paralleled to these connectors. These connectors are used to connect to a logic analyzer or a custom proto-type board.

***J7*** – Atmel microcontroller bus.

***J8*** – cPLD spare I/O pins.

***J9*** – 3.3V, 5V supplies. Used for monitoring the supply voltages or supplying the power to a custom proto-type board.

***J11*** - This connector is supplied for programming the cPLD (U1) on the S5335 PCI card. Either J11 or J12 may be used for programming the cPLD. This connector uses a non-flying leaded cable.

***J12*** - This connector is supplied for programming the cPLD (U1) on the S5335 PCI card. Either J11 or J12 may be used for programming the cPLD. This connector uses a flying leaded cable.

***J13*** – Atmel microcontroller programming pins.

***J14*** – RS-232 connector. This connector is used to connect to the PC's serial port. The connection is required to work properly with the AddOn windows application software, AMCCPCI5335AddOnWin. If AddOn side initiated bus mastering is not used, then this cable connection is not required.

***J15*** – nvRAM programming source. nvRAM may be set to program by either S5335 or Atmel microcontroller. The default jumper setting is programmed by S5335. The Atmel microcontroller software is not supported/supplied. The default jumpers are shorted between pin 1 to pin 3 and pin 2 to pin 4.

***J16 to J17*** - This is the primary PCI edge connector. All PCI communications and handshaking take place through this connector. Refer to the PCI SIG specification for signal names and definitions.

***J18*** - These two shunts located at JP16 are connected directly to the PCI edge connector. They provide the host platform with the logic states for the two PCI bus signals, PRSNT1# and PRSNT2# described in Section 4.4.1 of the PCI Local Bus Specification and are intended to report to the host the existence and the amount of power used by the PCI board.

| PRSNT1# | PRSNT2# | Definition |
|---|---|---|
| No shunt | No shunt | No expansion board present |
| Shunt | No shunt | Expansion board, 25W max |
| No shunt | Shunt | Expansion board, 15W max |
| Shunt | Shunt | Expansion board, 7.5W max |